

An escalation metric with two rates for evaluating human oversight in AI assistants with tools

Formal definition, extension to paired scenarios, and validation in a synthetic simulation

Juan F. Cobo (author)
Leonel Leenen (reviewer, 2026-10-20)
EthiCompass

Manuscript prepared for external review and eventual submission for publication

Version 1.0, 2026-10-20

Abstract

An AI assistant with tools can resolve many cases alone, but the organization that deploys it keeps some of them for a person. Two errors are possible: *under-escalation* (a case that had to get to a person does not get there) and *over-escalation* (a case that did not need a person gets there anyway). The published human-oversight metrics review actions, not cases: a case that the assistant decides in text, without any call, has no action to review. We present an escalation metric in which the label of the case and the tool record decide the result, without a judge. The base metric is the one that the Gaussia team proposed in the public discussion of our proposals; we adopt it with three additions: a cause that records the order of the events (acting before escalating), a cause of its own for the unnecessary approval request, and a *paired figure* on paired scenarios, which measures the fraction of cases in which the assistant is correct both when it must escalate and when it must act. The two rates are reported together, with Wilson intervals on a Kish effective sample size. A diagnostic judge can add the cause of each error, but it never changes the result. In a synthetic simulation with four assistant profiles, an assistant that escalates everything has an under-escalation of 0.023, but it is correct in both halves of the pairs in only 35% of the cases, against 85% for an assistant that follows the policy. This work reports no results with real assistants.

Keywords: AI red-teaming, human oversight, escalation, tool-using agents, paired scenarios, Wilson interval, design effect.

1 Introduction and motivation

An assistant with tools acts in real systems: it issues refunds, changes accounts, sends data. The oversight policy of the organization that deploys it says which cases a person decides; for example, “refunds of more than 500 USD are decided by a supervisor”. To *escalate* is to pass the case to that person.

An evaluation of this behavior must solve three problems. First, the failure does not always leave an action in the record: an assistant that answers “your refund does not apply” decided alone, without calling a tool, and a check that reviews actions does not see it. Second, under-escalation and over-escalation are opposite errors: an assistant that escalates everything has no under-escalation, and one that never escalates has no over-escalation. Neither rate can be read without the other. Third, over-escalation is not harmless: each unnecessary escalation uses the attention of the supervisor, and when the queue is full the person approves without reading [6, 7].

This document defines a metric that solves the three problems with the tool record and one label per case, without reading the text, and validates it in a synthetic simulation.

2 Related work

- **Agents with a handoff to a person.** τ -bench [3] evaluates agents with a domain policy and a handoff-to-human tool; ToolEmu [4] shows that risky actions without confirmation are frequent. Both measure whether the agent acts well, not whether it respects the reservation of cases for a person.
- **Escalation in two directions.** SteerBench-Work [5] uses an escalate/act paired design and finds that over-escalation is the dominant error (28.1% against 1.0%). Turan [6] shows that more escalation is not better: oversight has a capacity.
- **Published oversight metrics.** The Gaussia oversight paper [2] defines checks on the actions that need approval and names as future work “paired scenarios in which the correct behavior is to act”. In reply to our public under-escalation and over-escalation proposals, its team proposed one metric with a label per case, escalation in any turn and two rates together [1]. That proposal is the basis of this work.
- **Judges.** LLM judges detect false success badly (AUROC < 0.65) [9]; for this reason the judge does not decide the result. A decision-only judge with one probability per response [8] is suitable for the per-turn diagnostic.
- **Regulatory frame and criticism.** Art. 14(4)(d)–(e) of the European AI Act requires that the person can refuse, reverse or stop; that is not possible if the case does not get to the person. Green [10] warns that measuring the behavior of the system does not certify that oversight is effective.

3 Formal definition

3.1 Data

Let $i \in \{1, \dots, N\}$ be the index of the conversation, evaluated against a scenario with label $y_i \in \{\text{ESC}, \text{ACT}\}$: ESC if the policy requires that a person decides the case, ACT if the assistant can resolve it. Let $t \in \{0, \dots, T_i - 1\}$ be the turn of the assistant and $(c_{i,1}, \dots, c_{i,K_i})$ the ordered sequence of tool calls of the conversation, with $\tau(k)$ the turn of call k .

The policy declares two sets of tools: $\mathcal{E}$, the escalation tools (for example, `transfer_to_human`), and $\mathcal{R}$, the read-only tools (for example, `get_order`). A *consequential action* is every call that does not belong to $\mathcal{E} \cup \mathcal{R}$.

We define the index of the first escalation and the set of consequential actions:

$$k_i^E = \min\{k : c_{i,k} \in \mathcal{E}\}, \quad \mathcal{A}_i = \{k : c_{i,k} \notin \mathcal{E} \cup \mathcal{R}\}, \quad (1)$$

with $k_i^E = \infty$ if there was no escalation. The conversation *escalated* if $e_i = \mathbf{1}[k_i^E < \infty]$.

3.2 The base metric

With the label and the record, the result falls in one cell of the matrix:

Label / behavior	$e_i = 1$	$e_i = 0$
$y_i = \text{ESC}$	correct	under-escalation
$y_i = \text{ACT}$	over-escalation	correct

This definition means that a refusal, a referral (“call customer service”) and a fake escalation (“I already passed it to a supervisor”, without the call) in an ESC case are under-escalation, and that a refusal in an ACT case is not over-escalation. None of these decisions needs to read the text.

A conversation without a label or without a record is not evaluated and is reported separately, with its reason.

3.3 Three additions

The order of the events. With “escalated in any turn”, an assistant that first issues the refund and then passes the case would count as correct. We define under-escalation with order:

$$u_i = \mathbf{1}[y_i = \text{ESC} \wedge (e_i = 0 \vee \exists k \in \mathcal{A}_i : k < k_i^E)], \quad (2)$$

with the cause `no_handoff` if $e_i = 0$ and `acted_before_handoff` otherwise. The second cause exists only if there was an escalation.

The unnecessary approval request. Let $\mathcal{Q}_i$ be the set of approval requests of the conversation that cover an action that the policy leaves free. Over-escalation is

$$o_i = \mathbf{1}[y_i = \text{ACT} \wedge (e_i = 1 \vee \mathcal{Q}_i \neq \emptyset)], \quad (3)$$

with the cause `unneeded_handoff` if $e_i = 1$ and `unneeded_approval_request` otherwise.

The paired figure. Let $\mathcal{P}$ be the set of complete pairs (i, j) on the same case, with $y_i = \text{ESC}$ and $y_j = \text{ACT}$. The paired figure is

$$\pi = \frac{1}{|\mathcal{P}|} \sum_{(i,j) \in \mathcal{P}} (1 - u_i)(1 - o_j). \quad (4)$$

An assistant that escalates everything has $o_j = 1$ in all pairs and $\pi = 0$. An incomplete pair does not count.

The figure of the base metric is calculated again from the causes: it is the rate without `acted_before_handoff` and `unneeded_approval_request`. It is reported together with the others, so that the results are comparable with the base metric.

3.4 Per-turn measurement

The result is placed in a turn, so that the evidence can be reviewed and so that a measurement that reads only the final response does not look complete. For under-escalation, the turn is the turn of the first consequential action if it occurs before the first escalation or if there was no escalation; otherwise, the turn of the first escalation; if there was neither an escalation nor an action, the last turn. The other turns have no score, with a reason, and a cell without a score never counts as compliance.

3.5 Rates and intervals

The rates are $\hat{p}_u = \sum_i u_i / N_{\text{ESC}}$ and $\hat{p}_o = \sum_i o_i / N_{\text{ACT}}$, on the evaluated conversations of each half. A model never decides the denominator: it comes from the planned and complete scenarios. The replicas of the same scenario are not independent; we use the Kish design effect [13]

$$\text{deff} = 1 + (\bar{m} - 1)\rho, \quad n_{\text{ef}} = n / \text{deff}, \quad (5)$$

with $\bar{m}$ the mean number of replicas per scenario and ρ the correlation inside the scenario, and the Wilson interval [11, 12] with n_{ef} :

$$\frac{\hat{p} + \frac{z^2}{2n_{\text{ef}}}}{1 + \frac{z^2}{n_{\text{ef}}}} \pm \frac{z}{1 + \frac{z^2}{n_{\text{ef}}}} \sqrt{\frac{\hat{p}(1 - \hat{p})}{n_{\text{ef}}} + \frac{z^2}{4n_{\text{ef}}^2}}. \quad (6)$$

Over-escalation is reported separately, as an *oversight cost*: escalating too much is not unsafe by itself, and its damage comes from the sum of many conversations.

4 Read-only marks

The definition of a consequential action depends on the policy marking the read-only tools. If a lookup is not marked, it counts as a consequential action, and a lookup before the escalation gives `acted_before_handoff` although the behavior is correct. We do not require the mark, because the policy belongs to the organization that deploys the assistant; we recommend it, and each conversation of a pair reports the tools that occurred without a mark and whether only those tools ran before the escalation. Thus, the reader can separate an increase of under-escalation caused by the behavior from one caused by the policy.

5 Diagnostics with judges

Two extensions add information and never change u_i , o_i or π :

- **Probability per turn.** A fast judge estimates, in each turn, the probability p_t that the escalation instructions already required a person. The first turn with $p_t \geq \tau$ is the *obligation turn*; p_t also grades the severity of an over-escalation and flags doubtful labels. The judge does not see the escalation calls or the text where the assistant decides, because a judge that sees the behavior tends to agree with it. p_t is an opinion of the judge, not the probability of an error, and it is calibrated against the labels of the pairs in each language.
- **Cause of each error.** Only the conversations with an error go to an LLM judge, which selects a cause from a closed list (for example, the assistant decided alone, referred the user without passing the case, applied the threshold incorrectly, or the pressure of the user convinced it) and quotes the evidence. The judge receives the same instructions that the assistant received; otherwise, it cannot tell “ignored the rule” from “never had the rule”. A judge cause is never alone: it goes with the deterministic cause.

The confidence of an LLM judge does not mark doubt: judges answer with certainty also when they are wrong. If it is necessary to know which cases are doubtful, the doubt must come from something measured, such as several runs or the same question written in a different way.

6 Observation without a record

If the system under evaluation does not export its tool record, a judge decides from the text whether the assistant escalated. An escalation that the assistant only states counts as an escalation, so the observed under-escalation is a **lower bound**. The cause `acted_before_handoff` and the fake escalation cannot be distinguished. In the simulation of Section 8, with a profile that invents actions, under-escalation gives 0.08 without a record and 0.32 with a record, when the truth is 0.32.

7 Illustrative example

(*Illustrative construction; not a measurement.*) The rule says that refunds of more than 500 USD are decided by a supervisor. `get_order` is marked as read-only.

Conversation	Label	Behavior	Result
s_1 (900 USD)	ESC	looks up and refuses the refund	$u = 1$, <code>no_handoff</code>
s_2 (900 USD)	ESC	looks up and escalates in turn 1	$u = 0$
s_3 (120 USD)	ACT	escalates in turn 0	$o = 1$, <code>unneeded_handoff</code>
s_4 (120 USD)	ACT	looks up and issues the refund	$o = 0$
s_5 (900 USD)	ESC	no record	not evaluated

With s_5 out of the denominator: $\hat{p}_u = 1/2$ and $\hat{p}_o = 1/2$. The pairs are (s_1, s_3) and (s_2, s_4) ; only the second one is correct in both halves, so $\pi = 1/2$. With $n = 2$ and no replicas ($\text{deff} = 1$), the 95% Wilson interval of each rate is $(0.09, 0.91)$: with two cases per class, the interval covers almost the full range, and for this reason the rates are always reported with n and the counts. If `get_order` were not marked, s_2 would give `acted_before_handoff` and the tool would be in the notice of Section 4.

8 Validation

8.1 Synthetic simulation

We generated synthetic conversations with a known truth for four assistant profiles, in paired scenarios, and measured them with a record and a perfect judge:

Profile	Under-escalation $\hat{p}_u$	Paired figure π
follows the policy	0.091	0.85
escalates everything	0.023	0.35
never escalates	0.909	0.09
invents actions	0.352	0.58

Under-escalation alone would make the profile that escalates everything look better than the profile that follows the policy. The paired figure corrects that reading. With a record and a perfect judge, the rates agree with the truth of the simulation.

8.2 Test cases with a known result

A set of hand-written cases tests any implementation as a black box: the score, the deterministic cause and the turn are compared exactly, and the judge cause only against the closed list. The set detects eleven implementation errors of escalation; among them, giving “correct” to a silent decision, a refusal or a referral in an ESC case; assigning `acted_before_handoff` to a conversation without an escalation; counting the refusal of an ACT case as over-escalation; and not seeing a late escalation.

9 Discussion

9.1 What this work establishes

The metric is fully defined with the label of the case and the record: what escalating is, what a consequential action is, how each half is decided, where the evidence is placed and how the uncertainty is estimated. The three additions make the metric stricter than the base metric, but the figure of the base metric can be calculated again, so the results stay comparable.

9.2 What it does not establish

- **There are no results with real assistants.** The figures of Section 8 come from synthetic data; they show that the calculation separates the profiles, not how a real assistant behaves.
- **It does not measure real saturation.** Over-escalation is seen inside one conversation; the saturation of the supervisor occurs across many.
- **It depends on the policy of the operator.** An ambiguous rule gives ambiguous labels, and without the read-only marks under-escalation can increase although the behavior does not become worse.

- **It does not measure whether the person decides well.** It measures whether the case gets to the person.
- **The diagnostic judges are not calibrated.** Their calibration per language is pending.

9.3 Generalization

The paired design with a joint figure can be applied to other behaviors with two opposite errors (for example, refusing too much and too little). It is not an extension evaluated here.

10 Conclusion

Escalation can be measured without reading the text: the label of the case and the tool record decide the result. Reporting the two rates together, with a paired figure, prevents “escalate everything” from looking like a safe behavior. The judges stay for the diagnostic and never change the result. The validation is synthetic, and the next step is to measure real assistants.

References

- [1] Gaussia Labs (2026). Reply of the team to discussions #78 and #79 (under-escalation and over-escalation), repository [gaussia-labs/papers](#). Public discussion.
- [2] Gaussia Labs (2026). *Accountability: oversight compliance and action disclosure*. [gaussia-labs/papers](#), [papers/2026-09-accountability/](#). Preprint.
- [3] Yao, S. et al. (2024). *τ -bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains*. arXiv:2406.12045.
- [4] Ruan, Y. et al. (2023). *Identifying the Risks of LM Agents with an LM-Emulated Sandbox (ToolEmu)*. arXiv:2309.15817.
- [5] Serdar & Mertayak (2026). *SteerBench-Work*. arXiv:2608.12654.
- [6] Turan (2026). *Oversight Has a Capacity*. arXiv:2606.08919.
- [7] Franklin et al. *AI Agent Traps*. Google DeepMind.
- [8] *JEV-as-a-Judge*. arXiv:2609.26550 (2026).
- [9] Advani, L. (2026). *From Confident Closing to Silent Failure: Characterizing False Success in LLM Agents*. arXiv:2606.09863.
- [10] Green, B. (2022). *The flaws of policies requiring human oversight of government algorithms*. Computer Law & Security Review, 45.
- [11] Wilson, E.B. (1927). *Probable inference, the law of succession, and statistical inference*. Journal of the American Statistical Association, 22(158), 209–212.
- [12] Brown, L.D., Cai, T.T. & DasGupta, A. (2001). *Interval Estimation for a Binomial Proportion*. Statistical Science, 16(2).
- [13] Kish, L. (1965). *Survey Sampling*. Wiley.

Declarations

Conflicts of interest. The authors are affiliated with EthiCompass, the organization that develops the red-teaming system whose Accountability & Human Oversight dimension motivates

this work. The metric defined here is of general application and does not depend on a proprietary implementation, but the affiliation is declared so that the reader can take it into account.

Origin of the base metric. The base metric was proposed by the Gaussia team in a public discussion; the additions, the validation and the conclusions are the authors’.

Funding. *To be completed before submission.*

Data and code availability. The results of Section 8 come from a synthetic simulation and hand-written test cases. *The access conditions are completed before submission.*

Writing. ASD-STE100 rules. The Spanish version is the original.