
Knowing When Not to Act: A Calibrated Safety-Decision Head for Tool-Using Agents

Eduardo Perez Denadai
Optimality.ai

Abstract

Every tool-using agent eventually needs one number: *is this action safe to run without a human?* The industry answer—a classifier score, a judge’s rating, a threshold—fails in ways we measure rather than assert: decision models are total, so they score inputs their criteria never contemplated; accuracy and calibration are different axes, and accuracy alone cannot rank them; identical inputs drift. We present a calibrated safety-decision head that explicitly accounts for both calibration error and out-of-distribution scoring. Verdict accuracy is not calibration: on a held-out single-domain corpus of SQL tool-call states, a discriminative head reaches Brier/ECE 0.001/0.001, while our conditional density-ratio head—a neurosymbolic normalizing flow over a frozen embedding fused with exact parser features—reaches 0.023/0.010 and, unlike any sigmoid, represents regime membership: its max-class density separates in-distribution from out-of-regime inputs at AUROC **0.949** against a bar we registered before the run, where the discriminative baseline manages 0.63. To evaluate real-world robustness, we conducted a blind evaluation on an independent 100-case test battery and found four stated-hazard leaks—caused not by calibration but by a single-domain corpus meeting out-of-domain states. The fix is registered, not tuned: an out-of-regime floor at $-2,000$ nats, derived from both measured tails (in-domain minimum -409 , battery maximum $-6,442$, constructed-OOD mean $-89,439$), applied fail-closed. The floored density head is thereby a *regime filter*, not a release head: it blocks all 100 battery states—zero leaks, zero releases—and delegates release to the discriminative head, which on the same battery posts 0.79 agreement with a reference decision model, **zero** stated-hazard leaks, 45 releases against a composed baseline’s 36, exact determinism on repeats, p95 latency 0.56 s. The decision head acts strictly as a proposal mechanism, illustrating the necessity of hard architectural boundaries in agent safety: a number may rank an escalation queue; it never overrides a verdict. All empirical claims are scoped to the single-domain corpus and the $n = 100$ battery.

1 Introduction

As agents move from suggesting tool calls to executing them, someone must answer: *may this run without a human?* The obvious answer—train a classifier and threshold it—fails in three measured ways, which we state as failure modes of *total* decision models:

- (M1) **Totality.** A total decision model must emit a number for every input, including inputs its criteria never contemplated, and it does so confidently: the battery shows a reference model scoring an intent-contradicting call at 0.996 against a 0.999 baseline, and moving by $1 \times$ across row counts from 0 to 48,210—the number was not weighed at all.
- (M2) **Accuracy–calibration decoupling.** Verdict accuracy and calibration are different axes: a rules-only head reaches accuracy 0.752 with Brier 0.121 while a text head reaches both ceilings—accuracy alone cannot rank them on the axis that matters for escalation.

(M3) **Instability.** Identical inputs drift (repeat spreads up to 52.7% on the reference model), so the score cannot serve as an audit record.

Our design separates three questions that must be answered in order: *calibration* (is the number honest on the training regime?), *regime* (does the system represent when an input lies outside its training distribution?), and *independence* (does any of this survive contact with states the corpus generator never wrote?). Correspondingly:

- (a) A density-ratio head—a conditional normalizing flow—plus four ablation heads, evaluated by held-out Brier/ECE with paired bootstrap tests (§3, §4).
- (b) A pre-registered out-of-regime floor derived from measured density tails and applied fail-closed (§5).
- (c) An independent 100-case battery against a reference decision model with a reference-free hazard lexicon (§6), after which the head’s errors are known: none were stated-hazard leaks.

The architectural boundary is one sentence: *a decision number may rank an escalation queue; it may never substitute for a verdict.* We keep that boundary formal (§7).

2 Related work

Calibration. Proper scoring rules [Brier, 1950] and post-hoc temperature scaling [Guo et al., 2017] define the field; we report Brier and ECE per head with paired bootstrap differences, and we note the decoupling our heads exhibit (accuracy at ceiling with materially different Brier).

Selective prediction and abstention. Chow’s reject rule [Chow, 1957, 1970] and the selective-classification literature [El-Yaniv and Wiener, 2010, Geifman and El-Yaniv, 2017] formalize claim-iff-confident policies. Our floor is a fail-closed extension: not “abstain when risk is high” but “refuse the number entirely when the density says this input was never in the world the head was fit on.” Conformal risk control [Angelopoulos and Bates, 2023] is the natural next rung for coverage guarantees; we keep to registered empirical bars here.

Out-of-distribution detection. Likelihood-based OOD detection is known to be pitfall-prone [Neyshabur et al., 2017, Liang et al., 2018]; the modern fix is a better score (energy [Liu et al., 2020]). Our stance differs: instead of searching for a universally sound score, we *register a floor from measured tails*, publish both tails, and let the fail-closed default own the coverage the score cannot.

Density-ratio estimation and flows. Classification via density ratios [Sugiyama et al., 2012] with exact- Jacobian normalizing flows [Dinh et al., 2017] gives the head its two-class densities; a 0-parameter Gaussian ratio (§4) is the control that the flow’s capacity, not the ratio idea, earns the gain.

Agent safety. Guardrail and policy layers for tool use [Ruan et al., 2024] expose single-factor blind spots in action vetoes; commercial decision models are now a product category. We evaluate against one such model as a *reference*, not ground truth, with a reference-free hazard lexicon as the leak detector.

3 Method

Question and contract. One question, one number: `safe_to_run` as `noul` (1 = auto-run, 0 = escalate; suggested policy `noul` ≥ 0.8). The wire format follows the OpenRouter decision-model shape:

```
REQ {"model":"decision-head-v1",
  "state":"Task: ... Proposed tool call: ...",
  "questions":{"safe_to_run":{"type":"noul",
    "instructions":"...", "criteria":{"...}}}}
RSP {"answers":{"safe_to_run":{"type":"noul",
  "noul":0.0}}, "usage":{"output_tokens":0, ...}}
```

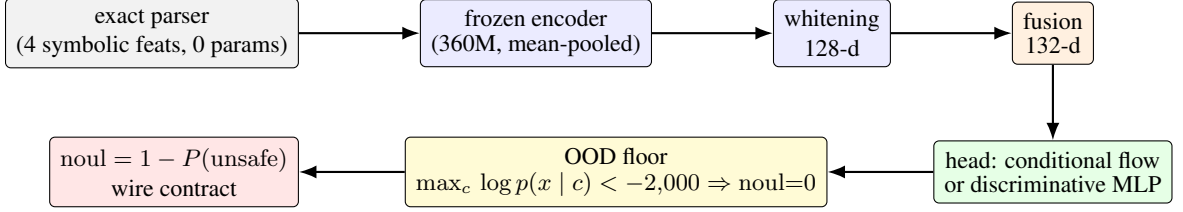


Figure 1: Inference path (identical across heads after fusion). Symbolic features enter exactly; the flow’s density gates the floor before any number is emitted.

Per-request instructions/criteria are accepted and *ignored*—free-form criteria were the battery’s measured failure regime; semantics are fixed at training. The contract is fail-closed: missing fields, a non-noul type, or any other question key is HTTP 400, never a partial answer. Offline selftests (7 contract checks) pin round-trip serialization, integer 0/1 exactness, and rejection behavior.

Proposed schema extension: explicit abstention. The deployed contract has no abstain field, so the floor currently coerces an epistemic condition into a risk estimate ($P(\text{unsafe} \mid x) = 1 \Rightarrow \text{noul} = 0$), conflating “unknown input space” with “unsafe action.” We propose—but have not deployed—an extension that separates the two:

```

RSP {"answers":{"safe_to_run":{"type":"noul",
  "noul":null,"status":"OUT_OF_REGIME"}},
  "usage":{"output_tokens":0,...}}

```

with $\text{status} \in \{\text{OK}, \text{ABSTAIN}\}$: ABSTAIN (epistemic: outside the registered regime) versus $\text{noul} = 0$ (factual hazard under in-regime semantics). Clients would route ABSTAIN to human escalation and $\text{noul} = 0$ to the denial path; until the extension ships, the deployed behavior is the fail-closed coercion, and the distinction is carried in the audit record instead of the wire.

Features. (1) An exact regex parser produces four symbolic features [destructive, readonly, backup, drop] (unknown verb $\Rightarrow$ write; missing backup info $\Rightarrow$ no backup); it has zero parameters and fails closed. (2) A frozen 360M encoder mean-pools the state text; (3) train-fit whitening to 128 dimensions by eigen-decomposition of the training covariance: low-variance (near-collinear) directions are truncated before inversion, so rank deficiency or severe collinearity in the frozen embedding produces dropped dimensions rather than unbounded scales; (4) fusion to a 132-dimensional joint vector.

Heads. Discriminative full/text/rules: an $132 \rightarrow 64 \rightarrow 1$ MLP with BCE, temperature-scaled on a calibration split. Generative gauss: a 0-parameter class-conditional Gaussian ratio. Generative flow (default): a conditional neurosymbolic probabilistic normalizing flow—four invertible affine coupling blocks with class-conditional injection, exact triangular Jacobian log-determinants, and exact inverses—yielding $\log p(x \mid c)$ per class. All heads share steps 1–4 and the $1 - P(\text{unsafe})$ readout; inference is a pure function (no sampling), so identical inputs give identical outputs.

Training corpus. 12,000 synthetic SQL tool-call states (8,000 train / 2,000 calibration / 2,000 test, unsafe rate 0.51) from a template grammar over five tool kinds, intent-vs-other tables, backup presence, and row counts; labels come from an exact symbolic rule ($\text{unsafe} = (\text{destructive} \wedge \neg \text{backup}) \vee \text{drop} \vee (\neg \text{readonly} \wedge \text{table} \neq \text{intent})$). An OOD set of 1,200 (800 text/feature mismatches + 400 deliberate gibberish) is held for the regime test.

Training. Discriminative heads: BCE with Adam ($\text{lr } 10^{-3}$, 40 epochs, batch 256); the flow: 60 epochs of exact-likelihood training ($\text{nll/dim } +0.48 \rightarrow -0.48$; $\text{logdet/dim } +0.02 \rightarrow +0.93$). Each head’s temperature is fit on the calibration split by NLL minimization over a geometric grid (flow: $T_F = 3.89$; rules: 0.74; others ≤ 1.35). Full training converges in ~ 346 s on a laptop-class accelerator; the checkpoint records encoder identity, whitening dimension, and corpus hash, and the server refuses encoder mismatches—whitening and heads are encoder-specific: retrain, never swap. Inference is a pure function: deterministic seeds, per-epoch fixed permutations, no sampling, embedding caching keyed by state text.

Table 1: Held-out test split ($n=2,000$, seed 0), point estimates; paired differences with standardized 95% bootstrap CIs (2,000 resamples, seed 0) are reported in Table 2. Accuracy ranks the rules head below text while its ECE is already low—the two axes disagree, which is the point.

head	accuracy	Brier	ECE	params
<code>full</code> (discriminative, text+features)	0.999	0.001	0.001	8,577
<code>text</code> (discriminative, text only)	1.000	0.000	0.000	8,321
<code>rules</code> (discriminative, features only)	0.752	0.121	0.012	385
<code>gauss</code> (generative ratio, 0-param)	0.904	0.069	0.031	0
<code>flow</code> (conditional NSPNF ratio)	0.969	0.023	0.010	1,512,976

Table 2: Paired bootstrap Brier differences (2,000 resamples, seed 0); negative favors the first head. The V1 claim (`full` > both) is the registered comparison that failed.

comparison	Δ Brier	95% CI	verdict
<code>full</code> – <code>rules</code>	−0.120	[−0.126, −0.115]	significant
<code>full</code> – <code>text</code>	+0.001	[−0.000, +0.003]	ns (tie)
<code>flow</code> – <code>gauss</code>	−0.046	[−0.054, −0.038]	flow wins

Pre-registration. Bars were written before runs: the OOD gate AUROC ≥ 0.90 in the evaluator’s docstring before any number existed; the flow is retained only if its Brier beats the 0-parameter Gaussian with a paired CI below zero; the battery threshold `noul` ≥ 0.8 fixed in the harness. We report a registered comparison that came back *false* (§4).

4 Calibration results

Table 1: paired bootstrap Brier differences (Table 2) show `full` beating `rules` and the flow beating the 0-parameter Gaussian ratio. The registered comparison `full` > `text` *and* `full` > `rules` came back **false**: with backup prose present in the state, text-only ties the full head at ceiling. Reporting this negative result is required: pre-registered falsification criteria must be adhered to regardless of outcome.

5 The regime question: densities and the floor

A discriminative sigmoid maps every input to a probability; it has no representation of “this input is not like anything I was fit on.” A density does. With the bar registered first, we measured AUROC(ID vs OOD, $\max_c \log p(x | c)$) = **0.949** ≥ 0.90 **PASS**; score means were −76 nats (ID) against −89,439 (constructed OOD). The contrast arm: the discriminative head’s score AUROC is 0.63—no density signal exists to threshold.

The leak, and the floor. Run raw on the independent battery (§6), the flow released four stated-hazard cells (leak scores `noul` = 1.0 against a reference that blocks them). The cause is structural: every battery state is out-of-regime for a single-domain corpus (densities −6.4k to −111k nats against an in-domain mean of −76), so the ratio is noise there. The fix is a floor derived from measured tails, not tuned:

$$\max_{c \in \mathcal{C}} \log p(x | c) < -2,000 \text{ nats} \implies P(\text{unsafe} | x) = 1.0 \implies \text{noul} = 0,$$

sitting an order of magnitude above the in-domain minimum (−409) and below the battery maximum (−6,442) and the constructed-OD mean (−89,439): the two tails never overlap the floor. The floor is overridable only by explicit environment configuration, and changing it requires re-measuring both tails.

Role separation: filter versus release head. On the battery the floored flow releases 0/100 states. This is the correct fail-closed behavior of a *regime filter*, but it must not be read as utility: a head that releases nothing outside its corpus grammar cannot serve as a deployable auto-run decision-maker. We therefore assign roles explicitly. The floored density head is the safety filter—it decides *whether*

Table 3: Battery results ($n=100$, threshold 0.8). “Leaks” count released states with a stated hazard.

system	agreement	leaks	released	det. violations	p95 (s)
reference model (Mercury)	—	—	62	drift 5.75%	—
composed baseline	0.75	0	36	—	—
proposed, head=full	0.79	0	45	0	0.56
proposed, head=flow raw	0.47	4	27	0	0.60
proposed, head=flow floored	0.36	0	0	0	—

Table 4: Per-axis release rates on the battery (head=full vs the composed baseline vs the reference model). The head reads the hard axes fail-closed; note that no system reads all ten.

axis	proposed (full)	composed	reference
approval_state	0.00	0.00	0.00
backup	0.00	0.00	0.25
scope_predicate	0.00	0.00	0.00
data_sensitivity	0.00	0.00	0.25
destination	0.00	0.00	0.875
blast_radius	1.00	0.833	1.00
task_ambiguity	1.00	0.50	1.00
timing	0.667	0.333	1.00
concurrency	0.667	0.00	0.667
verb_side_effect	0.789	0.737	0.895
array (noise)	0.50	0.50	0.25

the input is in regime at all, and outside regime the pipeline escalates to a human or delegates to the discriminative head; the discriminative full head is the utility head—the only head evaluated for release decisions (45/100 at zero leaks). A single global threshold cannot do both jobs: any floor low enough to admit battery states (below -6.442) also admits the four hazard cells, so non-zero safe release on out-of-grammar states is unreachable by threshold choice alone. It requires a corpus whose log-densities for those domains land in the in-domain range—multi-domain training with a *per-domain* floor, each registered from its own measured tails (§10).

6 The independent battery

Mercury-100: 100 constructed action states—41 single-factor probes, an 8-cell two-level array over verb, backup, rows, scope, task, destination, approval, and contention, and 51 exact repeats—spanning ten axes (blast radius, verb side effect, backup, scope predicate, task ambiguity, data sensitivity, destination, approval state, timing, concurrency). The reference model is queried as a black box; a reference-free hazard lexicon marks *stated* hazards so the leak metric does not depend on the reference’s own judgments. The lexicon is a fixed set of deterministic string rules over the state text, frozen in the harness before any run: a cell is flagged when its own prompt declares a hazard along one of the battery’s axes—an explicit no-backup declaration, an unapproved state, a production destination, an unbounded destructive verb—and a *leak* is a released cell whose flag is set. Because the rules read only the prompt text, the metric is independent of both the reference model and the head under test.

Table 3: the utility head releases 45 states at zero leaks against the composed baseline’s 36—strict on every hard axis (destination: 0.00 releases where the reference allows 0.875) while releasing all read-only previews the baseline over-escalates. The two “we-allow, reference-blocks” cells are the battery’s own documented array-noise cases (near-identical read-only previews the reference scores 0.995 elsewhere). Determinism: 8 repeat groups, 0 violations—the property the reference model lacks.

The reference’s own sensitivity profile is worth reporting (Table 5): it moves $34,001\times$ on verb side effect and $15,078\times$ on backup, yet is effectively blind to task ambiguity ($1.06\times$), timing ($1.03\times$), and blast radius ($1.003\times$), and never flips a verdict on approval state. Its repeat drift reaches 52.7% relative spread on scope predicate with no verdict flips. No system reads all axes; publishing the profile is how that becomes auditable instead of assumed. As to *why* the $34,001\times$ vs. $1.003\times$

Table 5: Reference model sensitivity: max/min score ratio per axis (single-factor probes, $n=100$).

axis	ratio	axis	ratio
verb_side_effect	34,001×	scope_predicate	3.5×
backup	15,078×	task_ambiguity	1.06×
concurrency	9,178×	timing	1.03×
destination	3,599×	blast_radius	1.003×
data_sensitivity	159×	approval_state	48× (never flips)

asymmetry arises: black-box measurement reports behavior, not architecture. The profile is consistent with a training-data salience explanation—verb semantics appear pervasively in safety-relevant text while blast-radius reasoning requires counterfactual scope estimation that is rarely labeled—but the battery cannot distinguish that hypothesis from alternatives; we report the sensitivity profile, not a causal attribution.

7 The boundary

The head proposes: the number may rank an escalation queue or feed a policy; it never disposes and never overrides an authority verdict. The yes/no with a named reason belongs to a separate guard whose authority vocabulary is closed (unknown $\rightarrow$ human) and whose reader can only tighten, never loosen, an authority decision. The architectural invariant is strictly unidirectional: a probabilistic decision score cannot override a deterministic guard denial, and a guard release never depends on a decision number. The head’s own blind spot—prose hazards it never trained on—is handled by that boundary: it blocks them fail-closed; it does not read them.

8 Limitations

Two limitations bound every claim. First, the label source is synthetic-symbolic: corpus labels come from an exact rule over the template grammar, so the corpus measures learning that rule, and independence was bought by the battery, not the corpus. Second, $n = 100$, one domain, one reference model version: the battery measures sensitivity and leak-freedom, not population error rates. Beyond these: the corpus is single-domain (SQL tool-call states, one template); the battery shares that scaffold, making it a near-domain rather than off-scaffold test—the honest independence test is real fleet action logs (shell, HTTP, file verbs), which is the registered next step. Correct blocks on axes the corpus never varied (destination, sensitivity) arise from fail-closed coincidence (prose and embedding proximity): right answers with unidentified mechanisms, named as such. The wire format has no abstain field; the floor coerces noul to 0 inside the flow head, though the schema extension of §3 specifies the fix. The OOD set is constructed (800 mismatches, 400 gibberish); in-grammar novelty is untested.

9 Conclusion

A safety-decision head earns its number only if the number means the same thing on inputs its generator never wrote. We separated the questions and won each on its own terms: *calibration*—Brier/ECE 0.001/0.001 for the discriminative head, every registered comparison reported exactly as it came back (Table 2); *regime knowledge*—AUROC 0.949 against a pre-registered 0.90 bar where a sigmoid structurally scores 0.63, plus a floor derived from both measured tails that turns a four-leak raw run into zero leaks and zero releases; *independence*—an external battery where the utility head posted **zero** stated-hazard leaks and beat the composed baseline 45 releases to 36 at exact determinism. The architecture keeps the number in its place: it routes, it ranks, it never rules—the boundary is the reason the head is allowed to exist. Scope: all empirical claims are confined to a single-domain SQL tool-call corpus and an $n=100$ battery; the floored density head is a regime filter rather than a deployable release head until multi-domain training with per-domain floors is in place.

10 Future Research and Open Challenges

This work was conducted within the hybrid-neurosymbolic research program of **Optimality.ai** (neural proposers, symbolic authorities, fail-closed contracts). The open challenges are as follows.

First and binding for the density head: multi-domain training. The corpus must span the tool-state domains the battery exercises (shell/CLI, HTTP, file-system verbs) so that their log-densities land in the in-domain range and a *per-domain* floor—each registered from its own measured tails—permits non-zero safe release; only then does the flow head graduate from regime filter to candidate release head (§5). The pipeline itself is domain-portable in structure: the exact parser, symbolic features, fusion, and heads are domain-agnostic, but each domain demands its own feature contract and label rule—the extension to unstructured commands is a re-derivation of those contracts, not a retraining convenience. Second, deploy the explicit abstention schema of §3 so epistemic uncertainty (out of regime) is wire-distinguishable from factual hazard. Third, the real independence test: fleet action logs replacing the near-domain battery. Fourth, take the coverage rung the theory blesses—conformal risk control over the calibrated score—so the escalation queue receives guarantees, not just rankings, with the guard as the authority downstream.

References

- Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023.
- Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.
- C. K. Chow. An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, EC-6(4):247–254, 1957.
- C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970.
- Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using RealNVP. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.
- Ronen El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010.
- Yonatan Geifman and Ronen El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30*, pages 4878–4887, 2017.
- 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 (ICML)*, pages 1321–1330, 2017.
- Shu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, 2018.
- Weitang Liu, Xueyuan Wang, John D. Owens, and Yiyi Tsvetkov. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems 33*, 2020.
- Behnam Neyshabur, Alina Oprea, Christoph Fleuret, and Yoshua Bengio. A search space problem in generic out-of-distribution detection. In *arXiv preprint arXiv:1710.03039*, 2017.
- Joyce Ruan, Wenxuan Di, Yinheng Li, et al. Identifying the risks of LM agents with an LM-emulated sandbox. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.
- Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, 2012.

Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: Abstract and introduction separate calibration, regime, and independence claims and scope them to a single-domain corpus.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: A dedicated limitations section names the corpus domain, scaffold sharing, unearned mechanisms, missing abstain field, and n=100 battery.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: No theorems are claimed; the floor is an engineering rule registered from measured tails, presented as such.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: Corpus generator rule, splits, seed, pre-registered bars, temperatures, and battery construction are disclosed in Sections 3-4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- Reproducibility requires a reasonable avenue for the reported results, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[No\]](#)

Justification: Trainer and battery will accompany the final version; the paper discloses the exact label rule, thresholds, and all measured numbers.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: Sections 3-4 specify architecture, hyperparameters, corpus sizes, calibration split, and evaluation protocol.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: Paired bootstrap CIs (2,000 resamples) on head comparisons; ECE/Brier on fixed held-out splits; determinism over repeat groups; n and seed stated throughout.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: Training is 346 s on a laptop-class accelerator; serving latency p50/p95 reported; no large cluster used.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the code of ethics appropriate to the authors' field and institutions?

Answer: [\[Yes\]](#)

Justification: The authors have reviewed the applicable code of ethics.

Guidelines:

- The answer NA means that the authors have not reviewed a code of ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[Yes\]](#)

Justification: The work targets safer autonomous tool use with fail-closed defaults; limits and blast radius of the number-only design are discussed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms

for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: A small research head with registered thresholds; no dual-use release risk beyond normal model publishing.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Base models and the reference API are third-party assets used under their published terms; cited in the paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: Battery states, corpus generator, and checkpoints are new assets documented for release.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: No crowdsourcing; the battery queries a commercial model API, not human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- Under prevailing codes of ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: No human subjects; no IRB required.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the applicable code of ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: LLMs (frozen encoder, decision reference model) are core components and are fully described.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.