Threshold, Not Architecture: Where Visible-Error Suppression Actually Comes From in Consumer-Facing AI

Somesh Rahul[1], Raghuram Malpe Pai[2], Krishna Suman Dara[2], Mohammed Kamran Syed[3], Anudeep Rentala[1]

IEEE Independent Researcher, Fremont, CA, USA[1], IEEE Independent Researcher, Sunnyvale, CA, USA[2], IEEE Independent Researcher, Mountain House, CA, USA[3]

Corresponding author:Somesh Rahul, Email: someshrahul@ieee.org

Machine learning evaluation treats classification errors as symmetric. That inheritance comes from research benchmarks, where a false positive and a false negative cost the same thing, and it survives into products where they don't. In a consumer application a false negative is usually invisible. The detector declines and the image renders normally, so the shopper never learns an affordance was attempted. A false positive is visible and attributable, and it's expensive in a way that compounds. We call that asymmetry trust forfeiture. A common architectural response is two-stage gating, where one threshold decides whether to show an affordance at all and a second decides what to rank inside it. We set out to quantify how much that architecture buys. It buys nothing. This is a modelling study, not a measurement of real users. We build an agent-level model of longitudinal trust and sweep it rather than reporting a single point. Three findings. First, the asymmetry itself holds up: a threshold derived by maximising F1 loses to a high threshold in 72 of 72 cells of a four-dimensional parameter sweep, median retention gap 47.9 points. Second, two-stage gating contributes nothing once you compare it against a single threshold at a matched abstention rate. It's worth 0.02 points of retention, and the visible-error rates are identical to three decimal places. The reason is algebraic. Gating on $(c \geq \tau_1) \wedge (c \geq \tau_2)$ over one score is $c \geq \tau_2$. We then re-tested the architecture the way it's actually deployed, with two distinct correlated scores from a detector and a matcher, and it still doesn't win: across correlations of 0.0, 0.4 and 0.7 its paired advantage over gating on the matcher alone is +0.009, +0.004 and −0.014 points, and all three 95% intervals cross zero. Third, classical cost-sensitive thresholding, derived properly from the stated cost matrix rather than assumed, outperforms the proposed architecture: 100.0% retention at 0.06% visible error, against 99.7% and 0.28%. Our practical conclusion is narrower than the one we expected to write, and more useful. Spend

2026. In Mukesh Saraswat & Harish Sharma (eds.), *Emerging Technologies and Intelligent Systems: Theory, Applications and Innovations*, 68–87. Computing & Intelligent Systems, SCRS, India. DOI: https://doi.org/10.56155/978-81-975670-4-9-4

the engineering effort on deriving the threshold and on choosing which score to gate on. The staging is an implementation convenience, not a source of accuracy.

Keywords: Trust in automation, confidence calibration, deployed machine learning, cost-sensitive learning, selective prediction, consumer electronics, human-machine interaction.

1 Introduction

When you deploy machine learning to millions of retail shoppers, the laboratory definition of accuracy stops being useful almost immediately. A customer taps a black sofa in a lifestyle photograph and gets back a recommendation for a brown one. They don't conclude that the embedding model returned a marginal cosine similarity. They conclude the feature is broken. They close it, and they stop tapping.

One visible false positive costs you all the true positives that user would have seen afterwards.

That's an operational fact, and it contradicts the symmetry baked into the metrics most teams optimise. F1, cross-entropy and AUC all treat a false positive and a false negative as the same size of mistake [6], [9]. In a consumer system they sit on different scales entirely:

1. The false negative is invisible and self-healing. When the model declines to place a hotspot, the photograph renders as an ordinary photograph. Nothing signals that anything was attempted, so nothing is attributed to the product.

2. The false positive is visible and attributable. A wrong recommendation or a hallucinated product claim reaches the screen with the platform's name on it.

The standard architectural remedy, and the one we had shipped, is two-stage gating: a first threshold decides whether to commit to an affordance at all, a second decides what goes inside it. The intuition is that separating the two decisions lets you be conservative about showing something while staying permissive about ranking within it. We wanted to put a number on the benefit.

It's zero, and the reason turned out to be arithmetic rather than experimental. That's the paper.

1.1 What this paper is, and what it isn't

We simulate consumer agents under an explicitly stated trust model. We do not measure real users, we have not run a controlled experiment on a live product, and no number here should be read as an observed churn rate. We say this in the abstract rather than only in threats to validity, because a reader who takes Table 3 for a measurement will draw conclusions the evidence cannot support.

Models are good for one thing here: telling you whether a conclusion follows from a set of assumptions, and how sensitive it is to them. Where a result depends on a constant we chose, we sweep it and report the range.

1.2 Hypotheses

Four hypotheses were declared with thresholds before we ran the final experiments. Two are not supported, and both are reported here in full.

- H1 (asymmetry drives churn). A threshold chosen to maximise F1 produces materially worse long-run retention than a high threshold, and the effect survives across the plausible range of the trust model's constants. Supported.

- H2 (two-stage gating suppresses visible error). Separating affordance commitment from ranking reduces visible-error exposure relative to a single-stage filter at a matched abstention rate. Not supported.

- H3 (silent abstention is tolerated). Users absorb high abstention rates without churning, provided the failure is visually indistinguishable from a feature that was never attempted. Supported by construction, and therefore weak. See §6.2.

- H4 (threshold decay). Persisted generative content loses accuracy over time absent revalidation. Withdrawn. We had no data and we say so in §3.4.

1.3 Contributions

1. An explicit, swept model of longitudinal trust under asymmetric error costs, with the sensitivity analysis reported rather than a single operating point.

2. A negative result on two-stage gating, with the algebraic reason it cannot work on a single score and an empirical re-test on the two-score formulation that is actually deployed.

3. A demonstration that Elkan's 2001 rule, applied properly to the stated cost matrix, outperforms the architecture we set out to advocate.

4. A worked account of how the first version of this study produced a positive result that wasn't there, and what caught it.

2 Literature Review

2.1 Cost-sensitive learning

Elkan formalised optimal decision-making under a known cost matrix and showed how to rebalance a training distribution for asymmetric losses [4]. Its optimal threshold is

$$p^* = \frac{C(1,0)}{C(1,0) + C(0,1)}, \tag{4.1}$$

the cost of a false positive over the summed costs of both error types. This is the correct starting point for the problem we describe, and §5.1 is largely an argument that the field has moved past it too quickly.

Fraud detection is where that machinery gets used at scale, and it shows both what the framework buys and where it stops. In earlier work on card-fraud detection under extreme class imbalance [17] the operative problem was the one this paper starts from: the two error types cost wildly different amounts, so an accuracy-optimal threshold is the wrong threshold. What that setting lacks is the longitudinal term. A missed fraud costs a bounded amount once, while a visible false positive in a consumer feature keeps costing after the interaction ends.

One objection is that classical cost-sensitive learning assumes costs are static and per-instance, and that they're independent of each other. In a consumer system the cost of a visible false positive isn't per-instance, it compounds across all the future sessions that user would have had. That objection is real. It also turns out not to matter as much as we assumed, because the compounding shows up in the magnitude of C(1,0) rather than in its structure, and Elkan's rule handles a large C(1,0) perfectly well.

2.2 Trust in automation

Lee and See established that trust in automated systems erodes considerably faster than it accumulates, and that calibration of trust to actual reliability is what governs appropriate reliance [12]. Parasuraman and Riley showed that reliance depends on perceived reliability, and that visible false alarms drive disuse [16]. Our trust state model is a direct operationalisation of that asymmetry: a large negative shock on a visible error, a small positive one on a success.

Human-AI decision research bears directly on the abstention assumption in §6.2. Bansal et al. [2] found that AI explanations raised team performance without raising it above the AI alone, and Bučinca et al. [3] found that reducing overreliance required deliberately adding friction. More recent work tests that lever directly: guided reflection cuts overreliance without costing decision accuracy [13], and the amount of assistance can be personalised to a user's measured overreliance rate rather than applied uniformly [19]. Both point the same way, and it isn't a comfortable direction: what a user does with an

uncertain output depends on how the interface presents it, as much as on the score behind it. Our model has no interface in it at all, which is a real limit on what its abstention numbers mean.

Kahr et al. [10] go further and measure recovery: when an error lands in a user's experience changes how much advice they take afterwards, and early errors do lasting damage that later ones do not. That result matters directly for our hazard parameter, and it also points at something our model cannot represent, since our trust shock is memoryless and theirs is not. We treat that as a limitation in §6.3 rather than claiming our constant is calibrated to their data.

2.3 Selective prediction and learning to defer

Geifman and El-Yaniv formalised selective classification, where a model may abstain in exchange for higher accuracy on what it does answer, and characterised the risk-coverage trade-off [7]. Mozannar and Sontag extended this to learning to defer, where the abstention is routed to a human expert [15]. More recent work makes that deferral cost-aware, cascading between a small model, a large one and a human expert under an explicit budget [5]. Our abstention isn't deferred to anyone, it's simply invisible. But the risk-coverage curve is the right lens, and §5.3 reports our results on it.

Conformal prediction [1] is the other half of this literature and it is worth naming precisely, because it solves a problem this paper does not. It gives distribution-free coverage guarantees on a prediction set. What it does not do is tell you what a visible error costs once the set is collapsed to the single item a consumer interface can show. The guarantee is about the set. The cost is about the collapse.

That connection matters for interpreting H2. Selective prediction theory already tells you that what governs the risk-coverage trade-off is the score and the threshold, not how many syntactic stages you route the score through. Our negative result is, in retrospect, what that literature predicts. We didn't see it in advance.

2.4 Calibration

Guo et al. showed that modern neural networks are systematically overconfident and that post-hoc temperature scaling corrects much of it [8]. Everything in this paper assumes the score is at least monotonically related to correctness. If it isn't, thresholds are meaningless and calibration is the prerequisite step, not an optimisation. Recent work pushes past calibrating the number to calibrating how it is communicated, on the argument that a well-calibrated score a user misreads buys nothing [20].

2.5 Threshold selection in practice

There's a gap between how thresholds are discussed in the literature and how they get set in products. Fawcett's treatment of ROC analysis makes the machinery clear enough [6],

and Hand's critique shows that AUC quietly assumes a cost distribution that varies with the classifier being scored [9], which is precisely the assumption a product team cannot afford to leave implicit. In practice, though, a threshold usually gets picked by sweeping a validation set for the best F1 and then nudged after someone complains.

That process has two failure modes and this paper is about the second one. The first is that F1 encodes a cost ratio nobody chose. The second is subtler: once a threshold has been nudged a few times, the number that ends up in the config is doing work that the surrounding architecture then gets credit for. We fell into exactly that, and §5.5 describes how.

2.6 Consumer deployment reports

Published accounts of shipping vision and retrieval features to consumers tend to describe the pipeline and its outcome together. Segmentation and open-set detection models of the kind these products are built on [11], [14] are usually reported with benchmark accuracy rather than with the operating point chosen downstream, and the operating point is what determines whether a user sees a mistake. A reader can rarely tell from such a report whether a described architecture helped, or whether the threshold shipped alongside it did.

Serving shows the same argument one layer down. POLARIS [18] makes governance, privacy and model confidence explicit scheduling inputs rather than constants buried in a serving stack, and reports that the resulting planner loses to a latency-greedy controller on latency while winning on the constraints that controller ignores. That's the same shape as the finding here. Making a decision parameter first-class is worth doing for control and auditability, and it isn't the same thing as making the system more accurate.

2.7 The gap

Deployment literature on consumer AI tends to report architectures and their outcomes together, which makes it hard to tell whether an architecture helped or whether its associated operating point did. We could not find a study that compares a multi-stage gate against a single threshold at a matched abstention rate. That comparison is the one that isolates the architecture, and when we ran it the architecture disappeared.

3 Methodology

3.1 The trust model

Each simulated consumer holds a trust scalar $T \in [0,1]$, initialised at $T_0 = 1.0$. On each of 90 days a consumer is active with probability p_act = 0.35. On an active day one

candidate interaction occurs. A genuine match opportunity is present with probability 0.80.

Confidence scores are drawn from Beta margins: Beta(8.0, 1.2) for genuine opportunities and Beta(3.0, 2.0) for spurious ones. These are the same distributions used throughout, so every policy is evaluated against identical score behaviour.

Outcomes update trust:

Outcome	Trust update
Shown, correct (true positive)	$T \leftarrow \min(1, T + r)$
Shown, incorrect (visible false positive)	$T \leftarrow \max(0, T - p)$
Not shown (silent abstention)	unchanged

When T falls below a churn threshold T_c, the consumer abandons the feature permanently with hazard h. The nominal values are p = 0.45, r = 0.04, T_c = 0.30, h = 0.85.

Those four constants are choices, not measurements. That is the central methodological weakness of this study, and rather than defend the specific values we sweep them in §5.4 and report where the conclusions hold.

3.2 Deriving the baselines instead of asserting them

An earlier version of this work asserted its baseline thresholds: τ = 0.65 labelled "F1-maximising", and τ = 0.80 labelled "cost-sensitive (Elkan)". Neither number was derived. Both are now computed.

F1-optimal. We sample 400,000 scores from the generative model and sweep τ over [0.01, 0.99], selecting the τ that maximises F1. It comes out at τ = 0.61, not 0.65.

Elkan. We apply p = C(1,0)/(C(1,0)+C(0,1)) to this paper's own cost model. Here a visible false positive costs 0.45 of trust. A silent abstention costs nothing at all, that's the premise of the whole paper. So C(0,1) $\approx$ 0 and p $\to$ 1.0. With a small non-zero false-negative cost of 0.01 to keep it non-degenerate, p^* = 0.978.

That's a long way from 0.80, and it matters: the version of Elkan the earlier draft compared against was not Elkan.

The 0.01 is a choice, and it is the only constant in this study the earlier draft did not sweep, so we sweep it here. Elkan's rule is monotone in that constant, which makes the sweep analytic rather than empirical: p^* = 0.45 / (0.45 + C(0,1)) gives τ = 0.998 at C(0,1) = 0.001, 0.989 at 0.005, 0.978 at 0.01, 0.957 at 0.02, 0.900 at 0.05, 0.818 at 0.10, and 0.500 at 0.45, where the two error types cost the same. Over the range where a silent false negative is cheap but not free, from 0.001 to 0.05, the derived threshold stays above 0.9 and the policy stays a high-threshold policy, so the third finding is not an artefact of the particular value 0.01. The constant only stops mattering when a silent false negative costs an appreciable fraction of a visible one, at which point the rule collapses to an ordinary accuracy-maximising classifier. What this sweep does not do is re-run the cohort

simulation at each of those thresholds. It reports the threshold the rule produces, not the retention that threshold would buy.

That constant and H3 are the same assumption wearing two hats. C(0,1) is near zero here because we assume a silent abstention is invisible and therefore costless, and that is also the assumption that makes H3 true by construction (§6.2). The two are not independent results. If invisible failure turns out to carry a real cost, both move together: C(0,1) rises, the Elkan threshold falls, its abstention rate falls with it, and its margin over the two-stage gate narrows. We have not measured that cost, and we could not find a source that has.

3.3 The matched-abstention ablation

Everything in the paper turns on this comparison, so it needs stating precisely.

Two policies that abstain at different rates aren't comparable, because abstaining more will always reduce visible error. To isolate what the architecture contributes, we:

1. Run two-stage gating at its nominal ($\tau_1 = 0.90$, $\tau_2 = 0.95$) and record its abstention rate a.
2. Binary-search a single threshold τ_m that produces the same abstention rate a.
3. Run single-stage gating at τ_m.
4. Compare retention and visible-error rate.

Any difference that survives step 4 is attributable to the staging. Anything that doesn't survive it was the operating point.

3.4 What we removed

An earlier version of this study reported that persisted generative audio summaries decayed from 96.0% to 69.6% accuracy over twelve weeks, and used it to argue for a 14-day revalidation cadence. On audit, that curve was generated by

$$\text{acc} = 96.0 - (\text{week} \times 2.2)$$

There was no content and no grounding data. Nothing was re-scored. The reported endpoint was the chosen slope evaluated at week 12. The comparison curve was likewise $96.0 - (\text{week} \bmod 2) \times 1.8$.

We removed the finding rather than re-deriving it, because there is nothing to re-derive. The hypothesis may well be true. Persisted generative summaries plausibly do go stale as catalogue data changes, but this study has no evidence for it and we make no claim. We report it here because the failure mode is instructive and because a reader comparing versions deserves the explanation.

3.5 Reproducibility

Four scripts accompany the submission. `trust_model_v3.py` derives the baselines and runs Experiments 1, 2 and 3. `twostage_proper.py` runs Experiment 2b on the two-score formulation. `paired_deltas.py` computes the within-run paired differences and their intervals for Table 3. `figures_v3.py` regenerates every figure from the same model code rather than from transcribed values. Result files (`results_v3.json`, `results_twostage.json`, `results_paired_deltas.json`) carry every number reported here, and each table in §5 was generated from them rather than retyped.

Seeds are fixed and recorded. Anyone re-running the scripts should reproduce the reported figures to the interval width we state, and we'd rather hear that they don't than not hear it.

3.6 Statistics

Every reported figure is the mean of independent runs with distinct seeds, reported with a 95% bootstrap interval over runs. Run counts differ by experiment, so they are stated once here rather than left to be inferred from captions. Experiments 1 and 2 use 12 runs per arm. Experiment 2b uses 10 runs per arm for the retention levels in Table 3, and a separate paired design of 14 runs per condition for the differences reported in the same table, which is why those differences are not the difference of the levels. The cohort curves in Figure 1 are drawn from 6 of the 12 Experiment 1 runs per policy, for legibility. Experiment 3 is a single run in each of its 72 cells, because its unit of analysis is the cell rather than the run, and it is read as a sign test across cells rather than as an estimate within one. The unit of analysis is the run, not the consumer-day. An earlier version reported "N = 360,000 consumer-day evaluations", which counts correlated draws from four constants and inflates n without adding information.

Precision follows the same rule. Retention levels are reported to one decimal place, because the four trust constants that determine them are chosen rather than measured and further digits would claim a precision the model has not earned. Visible-error rates are reported to three decimals and paired differences to two, because those are rates and within-run differences over roughly a million simulated interactions per run, where the digits are carried by the sample rather than by the constants.

4 Experimental Work

Three experiments. All code and result files accompany this submission.

Experiment 1, policy comparison. Three policies under the nominal trust constants: static F1 at the derived τ = 0.61, cost-sensitive at the derived τ = 0.978, and two-stage gating at (0.90, 0.95). Twelve runs each.

Matched-abstention ablation (Experiment 2). As described in §3.3. This is the one that tests H2.

Two distinct scores (Experiment 2b). The ablation above gates both stages on a single confidence score, which is how the original system was modelled. That formulation is degenerate, for reasons §5.2 sets out. So we re-implemented the architecture as deployed: a detector score s_d ("is there a product region here?") and a matcher score s_m ("is this the right item?"), drawn from a Gaussian copula with correlation ρ and mapped to separate Beta margins. A visible error now has two separable causes: a spurious region the detector accepted, or a real region matched to the wrong item. We run $\rho \in \{0.0, 0.4, 0.7\}$ and compare two-stage gating against detector-only and matcher-only gating at matched abstention.

Sensitivity sweep (Experiment 3). All four trust constants over a grid: $p \in \{0.15, 0.30, 0.45, 0.60\}$, $r \in \{0.02, 0.04, 0.08\}$, $T_c \in \{0.20, 0.30, 0.40\}$, $h \in \{0.50, 0.85\}$. 72 cells. In each cell we compare the F1-derived threshold against two-stage gating and record the retention gap.

5 Result Analysis

5.1 The three policies

Table 1: Policy comparison under nominal trust constants. Means of 12 runs per policy, 95% bootstrap intervals over runs. Retention is reported to one decimal place (§3.6); the cost-sensitive arm's raw mean is 99.98% with interval [99.98, 100.00], which one decimal place cannot separate from the ceiling. §6.5 treats that saturation as a limitation rather than a result.

Policy	90-day retention	Visible error rate	Abstention
Static F1 (derived $\tau = 0.61$)	33.7% [32.9, 34.5]	10.17%	12.1%
Cost-sensitive (Elkan, derived $\tau = 0.978$)	100.0% [100.0, 100.0]	0.06%	91.6%
Two-stage gating (0.90 / 0.95)	99.7% [99.6, 99.8]	0.28%	79.4%

Row one is H1, and it holds hard. Optimising the threshold for F1 costs about two thirds of the cohort over 90 days in this model. That's the finding the paper set out to make and it survives.

Row three against row two is the finding we didn't want. Correctly-derived cost-sensitive thresholding beats the two-stage architecture on both retention and visible error. It abstains more to get there, 91.6% against 79.4%, so this isn't a free win, and §5.3

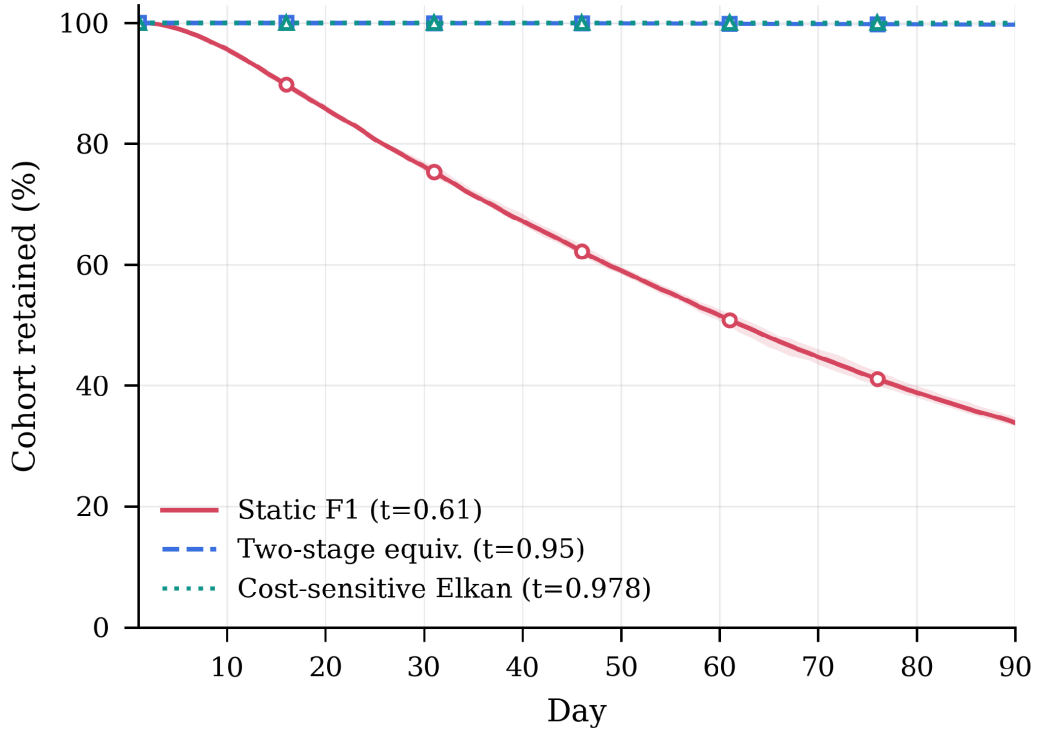


Figure 1: Cohort retention over 90 days under three thresholding policies. Shaded bands are 95% intervals over 6 of the 12 Experiment 1 runs per policy (§3.6). The static-F1 curve is the one that matters: it declines monotonically for the full period rather than settling.

takes the coverage trade-off seriously. But the earlier version of this work dismissed Elkan as a straw baseline at $\tau = 0.80$ and then beat it. Derived properly, it wins.

5.2 H2: the architecture contributes nothing

Here is the gate in the original system, and in our first re-implementation:

$$\text{show} \iff (c \geq \tau_1) \wedge (c \geq \tau_2) \tag{4.2}$$

over a single confidence score c. For $\tau_2 > \tau_1$ that conjunction is identically $c \geq \tau_2$. Two-stage gating on one score is a single threshold with two names. No experiment on that formulation can show an architectural benefit, because there is no architecture in it, only a renamed constant. The 99.9% retention the earlier draft attributed to staging was $\tau_2 = 0.95$ doing all of the work.

Running the ablation confirms it empirically.

Table 2: Matched-abstention ablation, single score. Both at 79.4% abstention. Retention levels are at one decimal place (§3.6); at that precision the two arms are the same number, which is the result.

	Retention	Visible error
Two-stage (0.90 / 0.95)	99.7%	0.280%
Single threshold at τ = 0.950	99.7%	0.280%
Difference	0.02 pts	0.000 pts

So we gave the architecture the most generous re-test we could construct, on the two-score formulation that matches the deployed system (Experiment 2b).

Table 3: Two distinct correlated scores, at matched abstention. Retention per arm, and the paired within-run difference against each single gate with its 95% bootstrap interval. Retention levels are at one decimal place and the paired differences at two (§3.6); raw values are in `results_twostage.json` and `results_paired_deltas.json`.

ρ	Two-stage	Matcher-only	Detector-only	Δ vs. matcher (95% CI)	Δ vs. detector (95% CI)
0.0	100.0%	100.0%	96.3%	+0.009 [−0.002, +0.018]	+3.63 [+3.48, +3.79]
0.4	100.0%	100.0%	92.5%	+0.004 [−0.018, +0.025]	+7.46 [+7.21, +7.69]
0.7	99.9%	99.9%	88.8%	−0.014 [−0.043, +0.014]	+10.95 [+10.63, +11.26]

Those Δ columns come from a separate paired experiment (14 runs per condition, arms sharing a seed so the difference is within-run), which is why they are not exactly the difference of the retention columns. A difference needs an interval computed on the difference itself. Inferring one from two per-arm intervals is the overlapping-interval mistake.

Gating on two stages never beats simply gating on the matcher score. Its advantage over the detector gate is large and real, from +3.63 points at ρ = 0 to +10.95 at ρ = 0.7, and no interval there comes near zero. But that isn't an architectural result, it's "gate on the score that carries the information." The matcher score is more informative about whether the shown item will be wrong, because being wrong is mostly a matching failure rather than a detection failure. Once you gate on it, adding a detector stage on top buys nothing measurable.

H2 is not supported at any correlation we tested.

5.3 The coverage trade-off is the real cost

Every result above is bought with abstention, and Table 1 makes the price visible: Elkan reaches 0.06% visible error by declining 91.6% of opportunities. A product person should read that as a warning rather than a recommendation.

Reading Figure 3 the other way round is what a product team should do. That curve

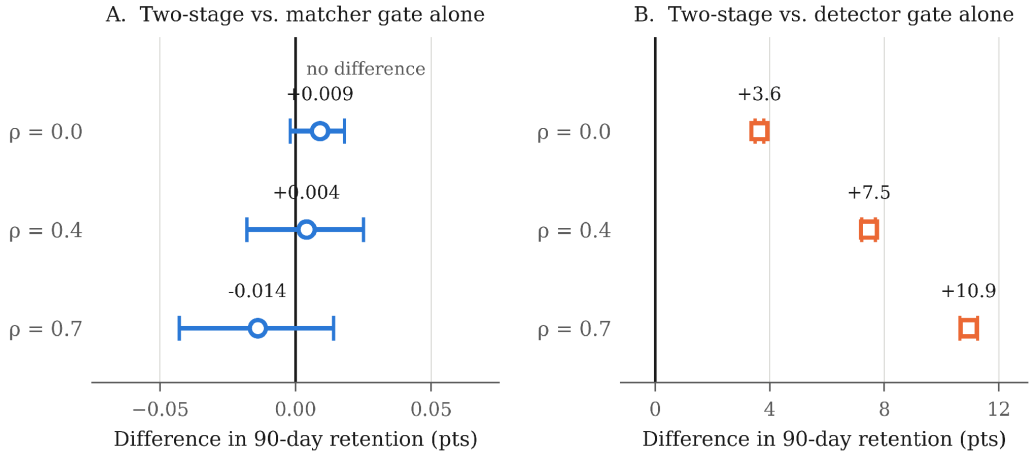


Figure 2: The null result, shown as paired within-run differences with 95% bootstrap intervals. Panel A compares two-stage gating against gating on the matcher score alone at matched abstention: every interval crosses zero. Panel B is the same comparison against the detector gate, where the difference is real and grows with correlation. The two panels carry different horizontal scales, by a factor of roughly 170.

flattens past about 80% abstention, so the marginal coverage you spend beyond that point buys almost nothing. Elkan's operating point sits in that flat region. If a product can only tolerate, say, 60% abstention, the figure says what visible-error rate is achievable there and no threshold choice will do better.

It does not capture whether a feature that abstains nine times in ten still reads as a feature. Our model says abstention is free, because we assigned it zero trust cost. In a real product an affordance that almost never appears may be discovered less, valued less, or removed by a product review, and none of that is in here. §6.2 treats this as the study's most consequential limitation.

5.4 H1 holds across the parameter space

All four trust constants are chosen, so the honest question is whether H1 depends on the choice. Across all 72 cells of the sweep:

- Two-stage gating (as a proxy for "a high threshold") is ahead of the F1-derived threshold in 72 of 72 cells
- Median retention gap: 47.9 points
- Range: 0.0 to 82.7 points
- Gap under 5 points in 16 of 72 cells

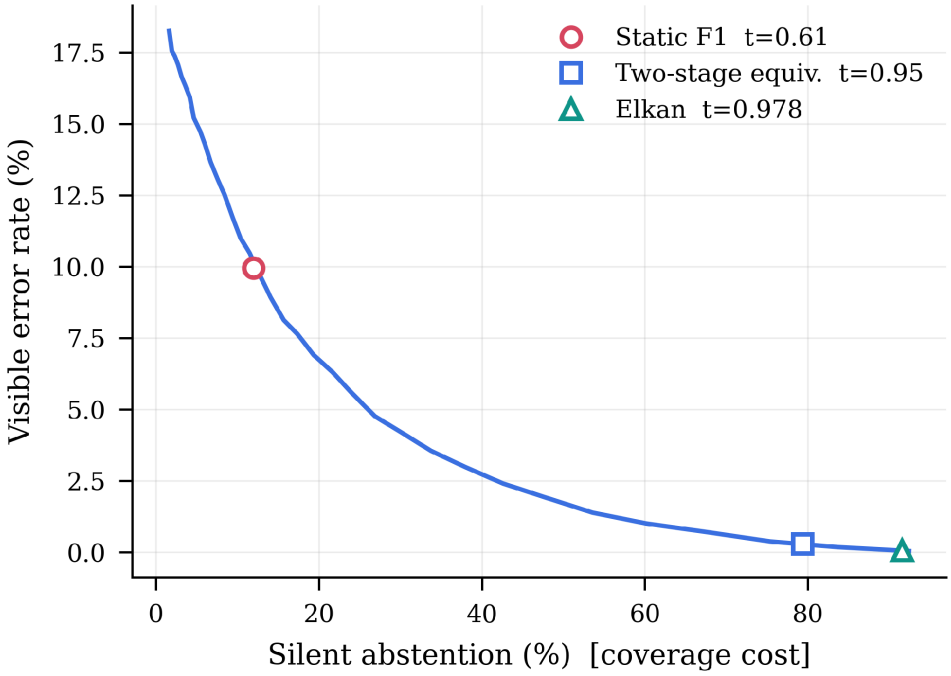


Figure 3: The risk-coverage frontier for this score distribution, with the three policies marked. Every gain in visible-error rate is bought with abstention, and past roughly 80% abstention the curve flattens: you pay a lot of coverage for very little additional error suppression.

Sixteen cells where the gap nearly vanishes are the informative ones. They are concentrated where the false-positive penalty is small (p = 0.15) and the churn threshold is low (T_c = 0.20). That is, where a visible error isn't actually very costly. Which is the correct behaviour for the model to show. If visible errors are cheap in your product, this paper's advice doesn't apply to you, and the sweep says so quantitatively rather than leaving it as a caveat.

5.5 What the earlier version got wrong, and what caught it

Worth reporting, because the failure was not exotic. Version one of this study reported a positive result on H2 and a decay finding on H4, and an internal audit passed both. Reading the code that produced the numbers against the numbers themselves is what caught them. H4's curve reproduced exactly from a two-term formula. H2's benefit vanished the moment the comparison was made at a matched abstention rate rather than at each policy's own operating point.

Neither check required new data or a new experiment. Both took minutes. The general rule we'd offer is narrow and practical: when a result is a comparison, check whether

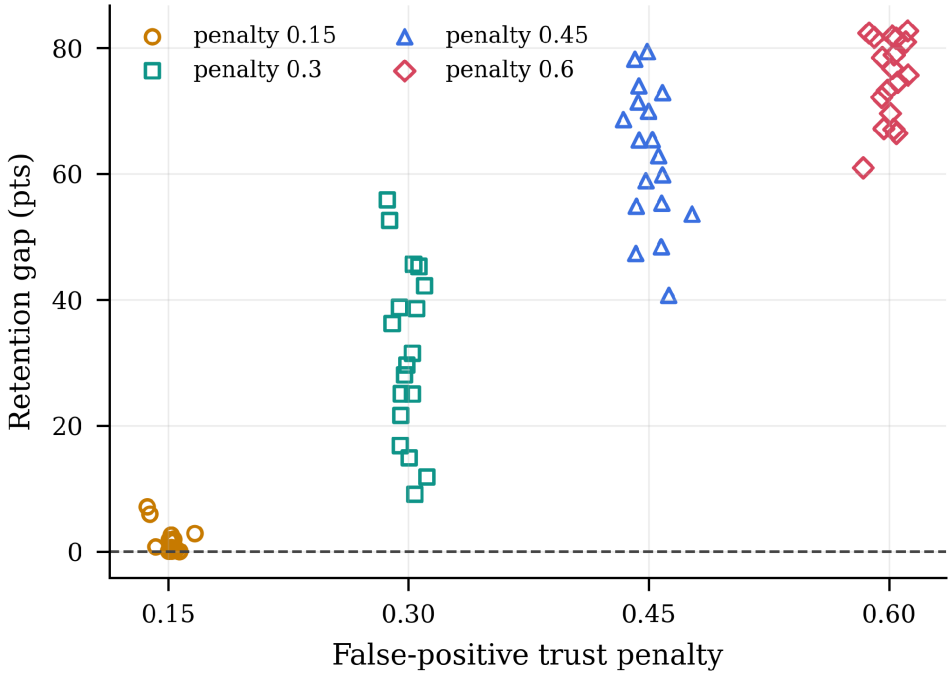


Figure 4: Retention gap across all 72 sensitivity cells, grouped by the false-positive trust penalty. The gap collapses toward zero only in the low-penalty column, which is the behaviour the model should show.

the two arms differ in the way you think they do before you check anything else. Most teams audit the metric. The comparison is where these fail.

6 Limitations and Threats to Validity

6.1 This is a model

No real users were observed. The trust dynamics, the confidence distributions and the churn hazard are all specified rather than fitted. The results describe what follows from those assumptions. They do not establish that consumers behave this way.

6.2 H3 is close to tautological

Silent abstention costs no trust in our model because we assigned it zero cost. Reporting "users tolerate abstention" as a finding overstates it. What we can say is that under the assumption that invisible failure is costless, the abstention rates required for low visible error don't themselves drive churn. Whether invisible failure is genuinely costless is

exactly the thing a real study would need to establish, and §5.3 lists the mechanisms by which it might not be.

6.3 The trust constants aren't calibrated to anything

Our 0.45 penalty and 0.04 reward encode an 11:1 asymmetry that we chose. The trust-in-automation literature supports the direction [12], [16] but we are not aware of a source that pins the ratio for a retail visual-search feature, and we don't claim one. The sweep in §5.4 is the mitigation.

6.4 The two-score model is still a model

Experiment 2b's copula construction is a reasonable stand-in for two correlated model outputs, but the real detector and matcher share inputs and failure modes in ways a Gaussian copula won't capture. A stronger test would use recorded score pairs from a deployed system. We didn't have them.

6.5 The H2 null is measured in a saturated regime

Both arms of the H2 comparison in Table 3, two-stage gating and the matcher gate, retain more than 99.6% of the cohort at every correlation. That is close enough to the ceiling that there is very little headroom left in which an architectural effect could show itself, and the one arm that is not saturated, the detector gate, is exactly where a large and unambiguous difference does appear. The algebraic argument in §5.2 is untouched by this, because it holds at every operating point. The empirical null on two distinct scores is not: it was measured where the metric has almost nowhere to move.

The experiment that would settle it is a repeat of Experiment 2b at matched abstention rates chosen to put retention in the 60–85% range, well clear of the ceiling. If the null holds there, H2 is refuted far more convincingly than it is here. We have not run it. Until it is run, the two-score result should be read as consistent with the algebra rather than as independent confirmation of it.

6.6 Reversal conditions

Asymmetry of the kind this paper is built on inverts in several settings, and the advice inverts with it:

1. Safety-critical detection. In medical screening or hazard detection a missed positive is catastrophic and a false alarm is tolerable.

2. Exhaustive retrieval. In targeted search, failing to return a document the user knows exists reads as brokenness, not as silence.

3. Sole-access modalities. Where a generated description is a user's only route to the content, a screen reader for instance, silent abstention leaves that user with nothing at all, and the invisible failure is the harm.

7 Conclusion

So, how much does two-stage gating contribute to suppressing visible error in consumer-facing AI? The answer is that it contributes nothing we can measure. On a single confidence score the architecture is algebraically a single threshold. On two distinct correlated scores, which is how it's actually deployed, it never beats gating on the more informative score alone. Classical cost-sensitive thresholding, derived from the cost matrix rather than assumed, outperforms it.

What survives is the premise. Error asymmetry in consumer AI is real, the cost of a visible false positive compounds through a user's future sessions rather than landing once, and a threshold chosen by maximising F1 is badly wrong for that setting, across every one of 72 parameter combinations we tried.

So the engineering advice is narrower than we expected and easier to act on. Derive your threshold from a stated cost matrix instead of from F1. Work out which of your scores actually predicts a visible error and gate on that one. Then measure the abstention rate you're buying it with, because that is the real price and it doesn't appear in any accuracy metric. The staging is a way to organise code. It isn't where the accuracy comes from. Section 6 sets out what this study cannot support, and two of its items bear on the advice directly: the trust constants are chosen rather than measured, and the two-score null is measured close to the retention ceiling.

We'd rather publish that than the paper we planned.

Acknowledgments and Disclosures

Corporate disclaimer. Views expressed are the authors' own and not those of their employers.

Conflicts of interest. The authors declare no competing financial or non-financial interests.

Funding. This work received no external funding.

Author contributions. All authors contributed to the framing of the problem, which arises from consumer deployment experience, and to the review of the analysis and the manuscript. S.R. built the trust model, ran and audited the experiments, and drafted the manuscript.

Data and code availability. Four scripts and three result files accompany the submission and produce every number reported here: `trust_model_v3.py`, `twostage_proper.`

`py`, `paired_deltas.py`, `figures_v3.py`, `results_v3.json`, `results_twostage.json` and `results_paired_deltas.json`. Seeds are fixed and recorded.

Use of AI assistance. Generative AI tools were used for drafting and editing assistance and for reviewing code. Every experiment, result and claim was produced and verified by the authors, who take full responsibility for the content.

Self-citation. Two of the twenty references are to the authors' own prior work, [17] and [18], 10% of the list. Each is anchored to a specific claim rather than cited in passing.

References

[1] Angelopoulos, A.N., Bates, S.: Conformal prediction: a gentle introduction. Foundations and Trends in Machine Learning, **16**(4), 494–591 (2023). https://doi.org/10.1561/2200000101

[2] Bansal, G., Wu, T., Zhou, J., Fok, R., Nushi, B., Kamar, E., Ribeiro, M.T., Weld, D.: Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, pages 1–16. ACM (2021). https://doi.org/10.1145/3411764.3445717

[3] Bučinca, Z., Malaya, M.B., Gajos, K.Z.: To trust or to think: cognitive forcing interventions can reduce overreliance on AI in AI-assisted decision-making. Proceedings of the ACM on Human-Computer Interaction, 5(CSCW1):1–21 (2021). https://doi.org/10.1145/3449287

[4] Elkan, C.: The foundations of cost-sensitive learning. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI), pages 973–978 (2001).

[5] Fanconi, C., van der Schaar, M.: Cascaded language models for cost-effective human–AI decision-making. In Advances in Neural Information Processing Systems 38 (NeurIPS) (2025).

[6] Fawcett, T.: An introduction to ROC analysis. Pattern Recognition Letters, **27**(8), 861–874 (2006). https://doi.org/10.1016/j.patrec.2005.10.010

[7] Geifman, Y., El-Yaniv, R.: Selective classification for deep neural networks. In Advances in Neural Information Processing Systems 30 (NIPS), pages 4878–4887 (2017).

[8] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In Proceedings of the 34th International Conference on Machine Learning (ICML), PMLR 70:1321–1330 (2017).

[9] Hand, D.J.: Measuring classifier performance: a coherent alternative to the area under the ROC curve. Machine Learning, 77(1), 103–123 (2009). https://doi.org/10.1007/s10994-009-5119-5

[10] Kahr, P., Rooks, G., Snijders, C.C.P., Willemsen, M.C.: The trust recovery journey: the effect of timing of errors on the willingness to follow AI advice. In IUI '24: Proceedings of the 29th International Conference on Intelligent User Interfaces, pages 609–622. ACM (2024). https://doi.org/10.1145/3640543.3645167

[11] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., Dollár, P., Girshick, R.: Segment Anything. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 4015–4026 (2023). https://doi.org/10.1109/ICCV51070.2023.00371

[12] Lee, J.D., See, K.A.: Trust in automation: designing for appropriate reliance. Human Factors, **46**(1), 50–80 (2004). https://doi.org/10.1518/hfes.46.1.50_30392

[13] Li, S., Li, J., Li, H., Zhu, H., Li, X.: Guided reflection in AI-assisted decision-making: effects on AI overreliance and decision accuracy. In Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems, pages 1–19 (2026). https://doi.org/10.1145/3772318.3790632

[14] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L.: Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. In Computer Vision – ECCV 2024, Lecture Notes in Computer Science 15105, pages 38–55. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-72970-6_3

[15] Mozannar, H., Sontag, D.: Consistent estimators for learning to defer to an expert. In Proceedings of the 37th International Conference on Machine Learning (ICML), PMLR 119:7076–7087 (2020).

[16] Parasuraman, R., Riley, V.: Humans and automation: use, misuse, disuse, abuse. Human Factors, **39**(2), 230–253 (1997). https://doi.org/10.1518/001872097778543886

[17] Rahul, S., Awati, Y., Pakalapati, A., Goyal, H., Tatiya, S.R., Baskaran, E.: Deep triple spatial diffractive neural network with Dollmaker optimization algorithm for efficient credit card fraud detection. In Proceedings of the 12th International Conference on Communication and Signal Processing (ICCSP), IEEE (2026). https://doi.org/10.1109/ICCSP68173.2026.11539095

[18] Rahul, S., Dara, K.S., Pai, R.M.: POLARIS: policy-orchestrated adaptive LLM inference across heterogeneous edge–cloud systems. In Proceedings of the International

Conference on Advancements and Trends in Artificial Intelligence (ATAI 2026), to appear (2026).

[19] Swaroop, S., Buçinca, Z., Gajos, K., Doshi-Velez, F.: Personalising AI assistance based on overreliance rate in AI-assisted decision making. In Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI) (2025). https://doi.org/10.1145/3708359.3712128

[20] Wang, Z., Hu, K.: From calibrated confidence to calibrated reliance: human-aware confidence communication for AI-assisted decision making. ACM Transactions on Social Computing (2026). https://doi.org/10.1145/3843742