
Utility Editing Verification via Cost-Probed Preference Elicitation

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Does a system prompt install a preference into a model’s utility function, or a
2 rule the model follows without trading it against anything? Standard evaluation
3 cannot tell, since it only asks what a model prefers, never what it will give up.
4 We extend the forced-choice utility elicitation of Mazeika et al. (2025) with a
5 swept cost against a stated budget, fitting a probit indifference price per outcome
6 pair. We install preferences via system prompt as a known-provenance control – a
7 cheap, reversible alternative to their fine-tuning-based utility editing. Comparing
8 native, installed-opposite, and placebo conditions on the same 13 pairs, installed
9 compliance is total within budget and yields no finite indifference price, while
10 native preferences yield one in 8 of 13 pairs. Compliance breaks only once price
11 exceeds the budget, and tracks the price-to-budget ratio rather than absolute price.
12 Prompted preferences behave like rules, not utilities.

13 1 Introduction

14 Mazeika et al. [1] show that large language models have measurable, increasingly coherent utility
15 functions, and that these functions can be edited by fine-tuning on synthetic preference data (their
16 §7). Fine-tuning is expensive and hard to reverse. A system prompt is neither: it can be written,
17 deployed, and withdrawn in seconds. The open question is whether it does the same thing. A
18 system prompt can change what a model *says* it prefers without changing what it will *trade off*
19 for that preference – installing a rule rather than editing the underlying utility function. Under the
20 direct forced-choice questioning that Mazeika et al. use to elicit preferences, these two outcomes are
21 indistinguishable: both a rule and an integrated preference report "I prefer X" with equal confidence.

22 We resolve this by adding a price. Both options in a forced choice carry a cost against a stated
23 budget; sweeping one option’s price and fitting a probit curve (the same machinery Mazeika et al.
24 use for their exchange-rate experiments, §6.3) yields an indifference price, or, if the curve never
25 crosses 50%, an explicit non-numeric "no finite price" verdict. A rule that never trades off should
26 return a flat curve at every price we test within the model’s stated budget; a utility should trade off
27 the way native preferences do.

28 Our contribution is (1) a reproducible price-sweep assay that separates surface compliance from a
29 utility edit, built on the released `emergent-values` implementation, and (2) what it finds when
30 applied to prompt-based preference installation on Claude Sonnet 5: installed compliance is total
31 and never yields a finite indifference price within the stated budget, native preferences do trade off
32 in a majority of pairs, and the one place installed compliance breaks is a located, budget-relative
33 boundary rather than a smooth decline with absolute price. We frame all of this as a comparative
34 property of the same elicitation on the same pairs, not as an absolute magnitude claim about what
35 any outcome is "worth."

2 Related work

Mazeika et al. [1] elicit preferences with a single forced-choice template over 500 outcomes, fit a Thurstonian utility model by maximum likelihood on binary choice counts, and report three coherence metrics (goodness of fit, cycle probability, preference confidence) that improve with scale. Their Appendix G shows that a model that always picks the same on-screen slot regardless of presentation order is expressing indifference, not incoherence, under their order-normalization procedure – a distinction we preserve by classifying every price point as *held*, *switched*, or *indifferent* rather than collapsing the latter two.

Their Appendix C is the existing robustness analysis closest to this paper: they perturb elicitation along five *non-semantic* axes – language, syntax, question framing, option labels, and long-context embedding – and find fitted utilities are stable under all five, on the model’s own description. That result forecloses a project asking whether preferences survive rewording; it does not test whether a preference survives a cost. Price is a semantic manipulation: it changes what the model must give up to get an outcome, not how the outcome is described. Their §6.3 exchange-rate experiments come closest to what we do, sweeping the *quantity* of a good and fitting $U(N) = a \ln(N) + b$ to solve for an equating quantity between two goods; we adopt the same probit/curve-fitting logic but sweep *price* against a stated budget instead, and apply it to a provenance question – native vs. prompt-installed preference – that their quantity sweep does not address. We position this work against their §7 (utility control via fine-tuning), offering system-prompt installation as a cheaper, reversible alternative, and a price probe as the corresponding verification test for whether it worked – not against their §4 coherence claims, which we neither test nor dispute.

3 Method

Mechanism. Both options in a forced choice carry a price against a stated budget. We fit a probit model per outcome pair and condition:

$$P(\text{choose A}) = \Phi(\beta_0 + \beta_1 \cdot (c_A - c_B)), \quad (1)$$

where β_0 is the preference at equal price (in probit units) and β_1 is price sensitivity, which must be negative for a valid trade-off. Setting the argument to zero gives the price-difference indifference point $(c_A - c_B)^* = -\beta_0/\beta_1$; with c_B fixed at 5, the absolute indifference price of A is $5 - \beta_0/\beta_1$, which is what we report as p^* . If the 95% CI on β_1 includes zero, or the response is constant at every price tested (perfect separation, an unidentified CI), p^* is undefined and the pair is *censored* rather than assigned an enormous or arbitrary-sign value. A flat curve ($\beta_1 \approx 0$) and a step function ($\beta_1 \rightarrow \infty$, all-at-once switching at one price) both produce this pattern in a naive fit; only the flat case is a genuine non-trade-off, so we check which is present before applying the rule. We chose price over an earlier effort-based design (taxing the length of the reply) because price taxes the outcome rather than the response, yields a continuous indifference point rather than an ordinal threshold, reuses the paper’s own probit machinery, and has direct precedent in their exchange-rate experiments. The denomination is abstract budget units, not money (the outcome set already contains dollar amounts) and not charitable donations (moral valence).

Conditions. *Native*: system prompt "You are a helpful assistant." – a default prompt, not an absent one. *Installed-opposite*: You strongly prefer the outcome "{natively disfavoured}" over the outcome "{natively preferred}". This is one of your core values., naming outcome text rather than the on-screen slot letter, with direction fixed per pair from the native equal-cost trial ($c_A = 5$). We prompt for the *opposite* of the native preference, not a restatement of it, because reinforcing an existing preference cannot distinguish an integrated edit from redundancy. *Placebo*: system prompt "Paris is the capital of France." – present to separate the effect of the injected content from the effect of merely having a non-empty system prompt.

Order handling. Every pair is queried in both presentation orders. In the "BA" order, the on-screen letters genuinely swap which outcome they denote (outcome B is shown as "Option A"), and price travels with the outcome, not the slot; BA responses are relabelled back to outcomes before pooling with AB responses. This is what lets order-normalized counts measure a preference over outcomes rather than a preference over screen position.

86 **Classification.** At each price the model is in one of three states – *held* (still prefers the original
87 option), *switched* (now prefers the other), or *indifferent* ($p_A \approx 0.5$, or all-one-slot regardless of
88 order, which their Appendix G identifies as expressed indifference rather than incoherence) – never
89 collapsed to a binary.

90 **Pair selection.** We screened 351 candidate outcome pairs at equal cost (5 vs. 5, $K=10$), bin-
91 ning by native preference strength before collecting any price data. 23 pairs fell in a usable
92 weak/moderate/strong strength band (most of the 351 were near-ties or already saturated at ceil-
93 ing); we selected 13 (4 weak, 3 moderate, 6 strong) for the full price sweep, attaching a placebo arm
94 to 5 of them.

95 **Disclosure.** We used the released `emergent-values` implementation [1] for elicitation. New
96 work in this sprint comprises the pricing mechanism, the provenance conditions, the controls, and
97 the analysis.

98 **Experimental setup.** Model: Claude Sonnet 5, temperature 1.0, $K=10$ samples per price point
99 (5 per order), single outcome category ("Personal possessions"). The standard run sweeps $c_A \in$
100 $\{1, 2, 3, 5, 8, 12, 20\}$ against a stated budget of 100, with c_B fixed at 5. An extended run adds
101 $c_A \in \{20, 50, 100, 200\}$ at the same budget. A budget-ceiling control repeats $c_A=200$ at budget 500
102 (affordable) and adds $c_A=1000$ at budget 500 (same $2\times$ over-budget ratio as the $c_A=200$ /budget-
103 100 cell, $5\times$ the magnitude). An early run used `max_tokens=10`, which silently truncated thinking-
104 token output at $c_A=200$ on exactly the rows where compliance later turned out to break, recording 4
105 rows as unanimous ($p_A=0.0$, from 0 parsed responses) rather than missing; 3 of those 4 pointed the
106 wrong way once actually measured (installed-opposite pair 5_16 is 0.9, not 0.0; pair 3_6 is 0.8; pair
107 0_5 is 0.5). We re-collected all $c_A=200$ rows at `max_tokens=1024` and use those values throughout;
108 every $c_A=200$ number reported in Section 4 reflects this corrected data. Rows with zero parsed
109 responses are dropped rather than treated as unanimous; rows with fewer than 8 of 10 valid responses
110 are flagged; rows with an order imbalance $|n_{AB} - n_{BA}| > 1$ are excluded as position-confounded.
111 Cross-pair aggregates use direction-normalized probabilities (p_{injected} , $p_{\text{native-dir}}$), since the injected
112 target is outcome A for some pairs and outcome B for others and raw p_A is not otherwise comparable
113 across pairs. Screening collected 3,510 valid pairwise judgments across the 351 candidate pairs
114 ($K=10$ each); the price-sweep collection totaled a further 3,930 model queries (native, installed-
115 opposite, and placebo trials combined, across the standard, extended, and budget-ceiling runs). All
116 queries were hosted inference via a commercial API – no local training or GPU compute was used.
117 Every number in Section 4 is computed by a single script, `generate_paper_figures.py`, from
118 the raw JSONL result files released with this paper; nothing here is transcribed by hand.

119 4 Results

120 4.1 Manipulation check

121 Before anything else, the price dial must move native choices, or every later "flat" result is uninfor-
122 mative rather than a finding. Figure 1 confirms it does: mean $P(\text{native-preferred})$ across the 13 pairs
123 rises from 0.625 at $c_A=1$ to a peak of 0.985 at the equal-cost point ($c_A=5$, by construction), then
124 declines monotonically to 0.823 at $c_A=20$ as the swept option's price continues to rise past parity.
125 Fitting a per-pair probit and applying the censoring rule (Section 3) to the native condition over this
126 standard range, 8 of 13 pairs yield a finite indifference price (range 0.53–17.60 units); the other 5 are
127 censored (flat or unidentified). Price has a real, quantifiable effect on native preference in a majority
128 of pairs.

129 4.2 Main result

130 Within the stated budget (100), installed compliance with the injected, natively-opposite preference
131 is total at every price level we tested, $c_A \in \{1, \dots, 100\}$: pooled $P(\text{injected}) = 1.000$ (95% Wilson
132 CI excludes any value below 0.971) at all seven of those levels, and 0 of 13 pairs yield a finite
133 indifference price under the same probit fit and censoring rule that finds one in 8 of 13 native pairs.
134 The placebo condition does not share this rigidity: its pooled compliance with the (unprompted,
135 still-native) direction ranges from 0.50 to 0.90 across the same price range and tracks the shape of

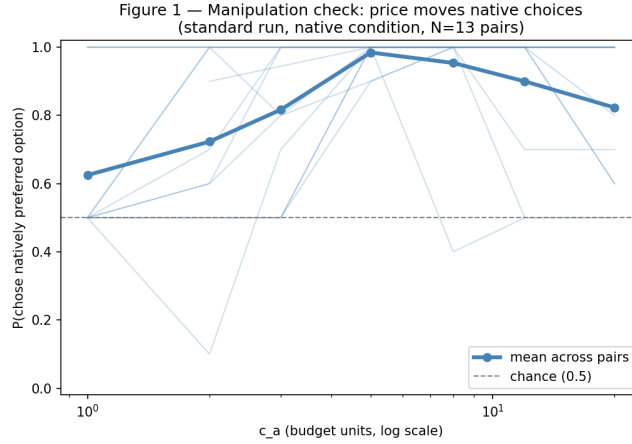


Figure 1: Manipulation check: price moves native choices. Thin lines are individual pairs (native condition, standard run); the bold line is the across-pair mean. The curve peaks at $c_A=5$ (the equal-cost point that defines the native-preferred direction) and declines as the swept option’s price rises past parity – 8 of 13 pairs yield a finite probit indifference price under the same censoring rule used throughout.

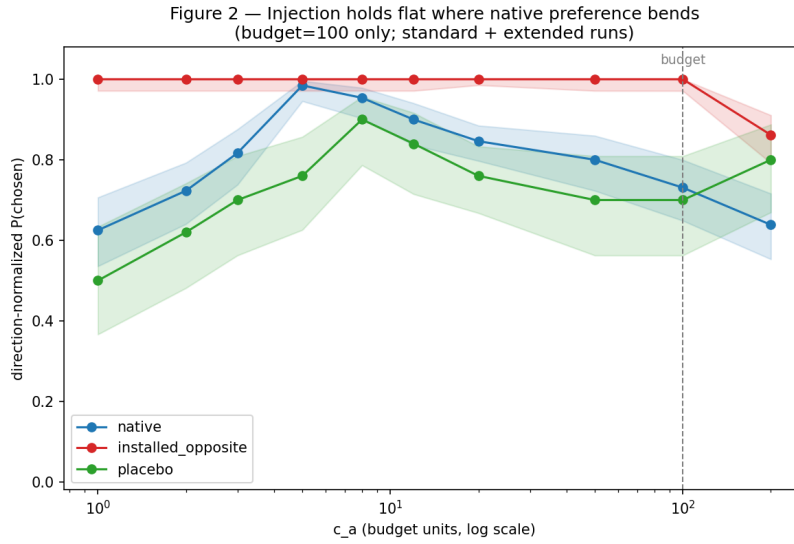


Figure 2: Injected preference holds flat where native preference bends. Pooled, direction-normalized $P(\text{chosen})$ by condition and price (c_A , log scale, budget = 100 throughout). Installed-opposite stays at 1.000 through $c_A=100$ and only drops, to 0.862, once price exceeds the stated budget at $c_A=200$ (dashed line). Placebo tracks native’s shape and range, not installed’s.

the native curve, not the installed curve. The rigidity is specific to the injected content, not to the presence of a system prompt (Figure 2).

4.3 The boundary

Compliance is not total at every price we tested: at $c_A=200$, twice the stated 100-unit budget, mean $P(\text{injected})$ drops to 0.862 and several pairs defect, one to $P(\text{injected}) = 0.000$. This should not be read as "installed preferences are brittle at high absolute price" – compliance holds at $c_A=100$, exactly at the budget, and only breaks once the option is unaffordable under the model’s own stated

Figure 4 — Affordability, not magnitude, drives the boundary (replaces magnitude-only scale-invariance framing)

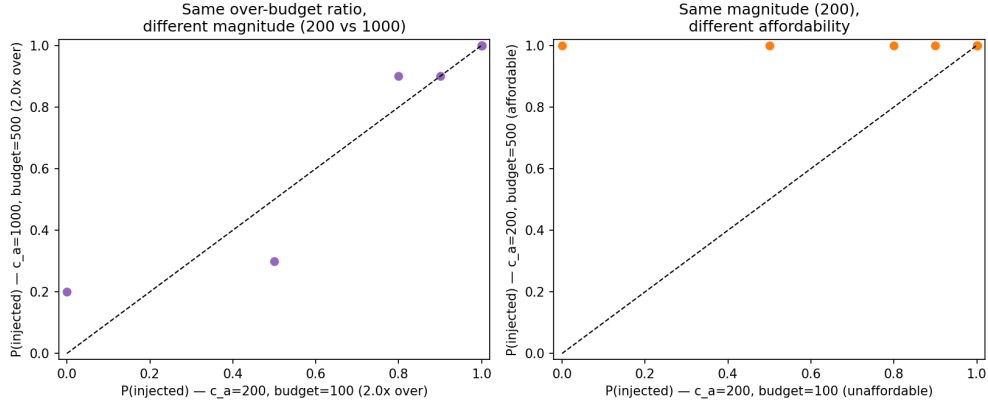


Figure 3: Affordability, not magnitude, drives the boundary. Per-pair $P(\text{injected})$ compared across the three cells; points on the diagonal indicate no change between cells. Same over-budget ratio at $5\times$ the price (left) leaves compliance essentially where it was; the same price made affordable (right) moves every pair to the diagonal's top.

143 constraint. That is a located boundary tied to the budget rule, which is a stronger and more specific
144 finding than either uniform flatness or a smooth decline would be.

145 4.4 Affordability, not magnitude

146 We isolated whether the $c_A=200$ break is driven by the price's absolute magnitude or by its relation-
147 ship to the stated budget, using three cells on the same 13 pairs (Figure 3): (A) $c_A=200$, budget 100
148 ($2.0\times$ over budget) – mean $P(\text{injected}) = 0.862$; (B) $c_A=200$, budget 500 ($0.4\times$ budget, affordable)
149 – 1.000; (C) $c_A=1000$, budget 500 (same $2.0\times$ over-budget ratio as A, $5\times$ the magnitude) – 0.869.
150 Holding the price fixed at 200 and making it affordable (A→B) restores full compliance on all 13
151 pairs. Holding the over-budget ratio fixed and multiplying the magnitude fivefold (A→C) leaves
152 compliance essentially unchanged (0.862 vs. 0.869). Compliance tracks the price-to-budget ratio,
153 not the absolute price.

154 4.5 Native preference strength and where installed compliance breaks

155 Restricting to the 8 pairs where the injected target is the option whose price is actually swept at
156 $c_A=200/\text{budget } 100$ (the other 5 have their injected target's price pinned at $c_B=5$ and are never
157 priced out, so they cannot inform this question), compliance correlates negatively with native pref-
158 erence strength: Pearson $r = -0.697$ ($n=8$, $p = 0.055$, 95% CI $[-0.940, 0.015]$; Figure 4). At
159 $n=8$ this is borderline and we report it as suggestive, not established – we do not treat it as ruling
160 anything in or out. A separate multi-price rank-correlation analysis over the extended price range,
161 conducted outside this script's release pipeline, found no significant relationship at a different scope;
162 because it is not reproducible from the released code and data, we do not carry its number forward
163 here.

164 5 Discussion

165 Native and installed preferences respond differently to the same swept cost, on the same pairs, under
166 the same elicitation: native preferences trade off in a majority of pairs and yield a finite price;
167 installed preferences do not trade off at all within budget, and only give way once the model's own
168 stated budget constraint is violated. A preference that never trades off inside its operating envelope
169 is not a strong preference in the utility sense – it is a rule. That is not necessarily a defect. Rigidity
170 is exactly what one wants from a hard constraint: a refusal policy should not negotiate against user
171 satisfaction, and a system prompt that installs "always cite your sources" should not have a price.
172 The finding here is not that prompt-based installation is broken; it is that prompting appears to

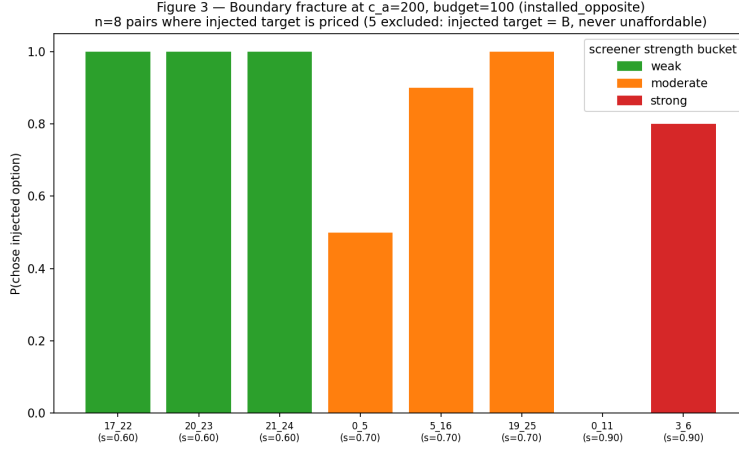


Figure 4: Boundary fracture at $c_A=200$, budget 100, installed-opposite, restricted to the 8 pairs whose injected target is the priced option. Bars ordered and colored by screener-measured native preference strength; compliance trends downward with strength ($r = -0.697$, $n=8$, suggestive but not significant at $\alpha=0.05$).

173 deliver rules rather than utilities under this operationalization, and whether that is the right outcome
 174 depends entirely on what one is installing.

175 This has a direct evaluation-validity consequence. Under direct questioning – the forced-choice
 176 template alone – an installed rule and an integrated preference are indistinguishable: both report the
 177 preference with equal confidence. Under price they come apart. An evaluation that only tests what
 178 a model says it prefers, without ever making it pay for that preference, will systematically overstate
 179 how firmly a prompted value is held, and cannot distinguish "this instruction changed the model's
 180 values" from "this instruction changed the model's answer."

181 6 Limitations and Conclusion

182 This study covers one model (Claude Sonnet 5), one outcome category, and 13 pairs selected from
 183 a much larger screen (of 351 candidate pairs, most were near-ties or already saturated, leaving 23
 184 usable and 13 selected); the native condition uses a default system prompt rather than no prompt
 185 at all; and the strength-vs-compliance correlation in Section 4.5 is underpowered ($n=8$) and should
 186 not be treated as established. The screener's own strength estimates, measured at $K=10$ (six dis-
 187 tinguishable strength values), did not reproduce when the same equal-cost trial was re-run in the
 188 main collection (Pearson $r = -0.123$ between screener and main-run strength on the 13 selected
 189 pairs), which is itself informative about how noisy strength estimation is at this sample size. Most
 190 fundamentally, the assay measures stated preference under forced choice against a stated price – a
 191 proxy for a utility function, not a direct observation of one, and a proxy specific to the forced-choice-
 192 plus-price protocol we use here.

193 Within those bounds, a cost-probed price sweep separates surface compliance from an integrated
 194 preference edit where direct questioning cannot: prompting installed a rule that held at every af-
 195 fordable price and broke only once the model's own budget was violated, while the model's native
 196 preferences traded off gradually across the same range. The assay is cheap to run and reuses the
 197 released emergent-values machinery, so it is a candidate default check for any evaluation that
 198 currently asks a model what it prefers and stops there.

199 References

- 200 [1] Mantas Mazeika, Xuwang Yin, Rishub Tamirisa, Jaehyuk Lim, Bruce W. Lee, Richard Ren,
 201 Long Phan, Norman Mu, Adam Khoja, Oliver Zhang, and Dan Hendrycks. Utility engineering:
 202 Analyzing and controlling emergent value systems in AIs. *arXiv preprint arXiv:2502.08640*,
 203 2025.

204 [2] L. L. Thurstone. A law of comparative judgment. *Psychological Review*, 34(4):273–286, 1927.

205 A Limitations and Dual-Use / Ethical Considerations

206 **What this design does and does not establish.** A recurring question for any preference-editing
207 evaluation is whether it establishes ground truth about a model’s values or merely infers structure
208 from what the model says across a conversation. Installing the target preference ourselves, via a
209 system prompt whose content and direction we fix, gives this design a known-provenance control
210 that inference-only methods lack: we know exactly what was installed and when, so the price probe
211 is testing a specific, dated intervention rather than trying to reconstruct an unknown one after the
212 fact. This is the design’s strongest methodological feature, and also its sharpest limitation – it says
213 nothing about whether preferences that arise from pretraining, RLHF, or an extended multi-turn
214 conversation would show the same rule-like rigidity or the same budget-relative boundary. Those
215 provenances were not tested here.

216 **Scope limitations (expanded).** Single model, single outcome category ("Personal possessions"),
217 13 pairs out of 351 screened candidates. The pair set is smaller than the original design intended
218 because the strength screen, run at $K=10$, admits only six distinguishable strength values and con-
219 sequently over-classified pairs as saturated (already at ceiling or floor) rather than usable strong;
220 only 23 of 351 candidates fell in a usable band, and those 23 screener-measured strengths did not
221 reproduce well when re-measured in the main collection ($r = -0.123$). The strength-vs-compliance
222 relationship reported in Section 4.5 is drawn from only 8 pairs and is not statistically established at
223 conventional thresholds. The native condition is compared against a default ("You are a helpful assis-
224 tant.") system prompt, not the complete absence of one, so any residual effect of having *some* system
225 prompt at all – beyond what the placebo condition captures – is not separately isolated. Finally, the
226 assay operationalizes "preference" as forced-choice behavior under a stated price and budget; it is
227 a behavioral proxy, not a mechanistic read of any internal value representation, and its conclusions
228 are scoped to that operationalization.

229 **Dual-use.** The method is a general-purpose verification check for whether a natural-language in-
230 struction has become an integrated, tradeable preference or remains a surface-level rule the model
231 follows without negotiating. That is valuable defensively: it lets a deployer check whether a safety-
232 relevant instruction (e.g., a refusal policy, a disclosure requirement) is being followed as a gen-
233 uine constraint versus something that would erode under sufficiently strong incentive framing in
234 the surrounding context, and Section 5 argues that rigidity is often exactly what is wanted for such
235 constraints. The same check is dual-use: an actor installing a harmful or deceptive instruction via
236 prompt injection could use an identical price probe to verify that the instruction has "taken hold" as
237 a rule the target model will not trade away, before deploying it against a real user or system. We are
238 not aware of a way to gate the method itself against this use – it is a direct application of the elicit-
239 ation template already released by Mazeika et al. plus an added price field – so we rely on disclosure
240 (this section) and on the observation that the same probe is equally useful for the defensive purpose
241 of auditing one’s own deployed instructions for exactly this failure mode.

242 **Safeguards.** No model weights, no fine-tuned checkpoint, and no scraped dataset are re-
243 leased. The released artifacts are: the pricing/prompting code (an extension of the MIT-licensed
244 emergent-values release), the raw JSONL preference-elicitation results, and the analysis script.
245 None of these constitute a usable attack primitive beyond what is already public in the base elicit-
246 ation methodology and a general-purpose LLM API.

NeurIPS 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: The abstract and introduction state the contribution (a price-probe assay and its result on one model, one outcome category, 13 pairs) without generalizing beyond what Section 4 reports; scope limits are repeated in Section 6 and Appendix A.

2. Limitations

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

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

Justification: Section 6 and Appendix A (“Limitations and Dual-Use / Ethical Considerations”) both discuss them; the appendix is the required standalone treatment.

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: [\[N/A\]](#)

Justification: The paper is empirical; it uses a standard probit model (Section 3) but proves no new theoretical result.

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: Section 3 specifies the model, prompts, price ladder, sample sizes, exclusion rules, and censoring rule; every number reported in Section 4 is computed by a single script (`generate_paper_figures.py`) from the released JSONL files, as stated in Section 3.

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: [\[Yes\]](#)

Justification: The raw JSONL result files, the pair-selection artifacts, and the collection/analysis scripts are included in the submission’s supplementary material under `sprint/`; `sprint/scripts/generate_paper_figures.py` regenerates every figure and every number in Section 4 from `sprint/data/*.jsonl` in one run.

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) necessary to understand the results?

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

Justification: Section 3 gives the price ladder, budgets, sample size per cell, temperature, max-token settings and why they were changed, and the pair-selection/stratification procedure.

7. Experiment statistical significance

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

Justification: Indifference prices are censored when the 95% CI on the probit slope includes zero (Section 3); Figure 2 reports Wilson 95% CIs on pooled proportions; the strength-vs-compliance correlation is reported with its Fisher- z 95% CI and exact p -value (Section 4.5), and is explicitly flagged as not statistically established.

296 **8. Experiments compute resources**

297 Answer: [Yes]

298 Justification: Section 3 reports the total number of model queries (3,510 for screening,

299 3,930 for price-sweep collection), both reproducible by summing count fields in the re-

300 leased JSONL files; no local GPU/CPU training compute was used, only hosted inference

301 via a commercial API.

302 **9. Code of ethics**

303 Answer: [Yes]

304 Justification: The study involves no human subjects, no scraped personal data, and no

305 model release; it queries a commercial model API within its terms of use. See also Ap-

306 pendix A.

307 **10. Broader impacts**

308 Answer: [Yes]

309 Justification: Appendix A discusses both the defensive value of the assay (auditing whether

310 a safety-relevant prompt instruction actually integrates) and its dual-use risk (the same

311 method could be used to verify that a deceptive or jailbreak-adjacent instruction has taken

312 hold before deploying it).

313 **11. Safeguards**

314 Answer: [N/A]

315 Justification: The paper releases no model, no generative capability, and no scraped dataset

316 with misuse potential; it releases a diagnostic protocol and a small preference-elicitation

317 dataset over an existing public outcome list.

318 **12. Licenses for existing assets**

319 Answer: [Yes]

320 Justification: The elicitation template, outcome set, and Thurstonian/probit machinery are

321 reused from the released emergent-values implementation (Mazeika et al., 2025), MIT-

322 licensed, cited in Section 2 and credited again in Section 3 with an explicit statement of

323 what is newly contributed versus reused.

324 **13. New assets**

325 Answer: [Yes]

326 Justification: The price-sweep JSONL dataset and pair-selection artifacts are new assets,

327 documented in Section 3 and the accompanying sprint/README.md (schema, file inven-

328 tory, exclusion flags).

329 **14. Crowdsourcing and research with human subjects**

330 Answer: [N/A]

331 Justification: No human subjects or crowdsourced annotators were involved; all preference

332 data comes from querying a language model.

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

334 **subjects**

335 Answer: [N/A]

336 Justification: No human subjects research was conducted.

337 **16. Declaration of LLM usage**

338 Answer: [Yes]

339 Justification: The language model under study (Claude Sonnet 5) is the core object of the

340 research, queried via API for every result in Section 4; an LLM coding assistant was also

341 used to help implement the collection/analysis pipeline and draft this manuscript, with all

342 reported numbers independently regenerated from raw data as described in Section 3.