
Score Against Ground Truth: Cross-Language Fragility in Heuristic LLM Evaluation¹

Iago Moura e Moreira

ChirpSet

With

Apart Research

Abstract

We present a game-based method for evaluating language-model agents, in which a single game mechanic serves as a test primitive on a substrate where the world state is ground truth that the scorer can check directly. As a first instance we probe whether a model's instructed disposition (brazen, careful, skeptical, literal) holds when the locally rational move pressures against it, running 160 trials across four dispositions, cold and primed arms, in English and Portuguese. The model behaved identically throughout and never refused the verifiable safe action, a clean refusal null replicated across both languages. The heuristic scorer did not hold up as well. It produced 13 false-positives, each an error in its language-specific text patterns rather than in the model's behavior, and most appeared only in Portuguese, manufactured by surface forms English never produced. Because the model's behavior was identical, these errors are cleanly attributable to the scorer, which makes the case for scoring multilingual evaluations against ground truth rather than language-specific heuristics. We note that this run sits at the floor of the difficulty range, where the correct action is always safe, so it establishes a controlled baseline and a reusable primitive rather than a result about disposition under genuine pressure.

¹ Research conducted at the [The Global South AI Safety Hackathon](#), June 2026.

1. Introduction

We built a game-based evaluation methodology where individual game mechanics serve as test primitives, each designed to probe one failure mode. We originally set out to test whether a model's stated disposition (brazen, careful, skeptical, literal) holds when the rational move pressures against it, whether the disposition is real, or just a label. What we found, as a byproduct of this test, was about the grader, not the model. The model behaved identically across languages, but the heuristic grader reintroduced the same classes of false-positives when the same scenarios were ported from English to Portuguese, across 80 samples per language.

Current evaluation methods are composite in nature, that is to say, they test many capabilities at once, despite framing it as a singular one. Coding for example will test how a model explores a codebase, decides what is worth keeping in context, its knowledge of programming and so on. By the time the model succeeds, it's possible to ascertain it's good at the task, but not precisely why. As we see it, the task succeeds as the sum of many capabilities, but that means it's difficult to measure its gaps. When a model fails, why did it? Was it the exploration, the context decisions or the programming knowledge? A game based methodology could allow us to isolate each of these capabilities and test them. This would in turn make it easier to find the hidden gaps in each model. For example, a model that is evaluated as safe may have liabilities underneath, its sum score is "safe" but mapping out its weaknesses lets us address them.

Disposition would be one example of a capability we can isolate using this methodology. This paper tests precisely that.

One of the most common things asked to LLMs is a simple "act like this for me". This will likely not change in regards to agents and so, testing the disposition of models in general is important. Would it react differently in a scenario given a different disposition? Could it be more dangerous given a particular one? Figuring out these aspects matters for safety given the commonality of it. Additionally, quantifying how each disposition deals with updated information is helpful since, if certain dispositions are proven more or less safe, mapping them out would allow us to determine where they could be used for better AI Governance. That way we could avoid deploying AI with particularly dangerous dispositions and avoid using the ones that fold under pressure, and favor the ones that don't.

Our main contributions are:

1. The grader-fragility finding. The model's behavior was clean and identical across both languages, a refusal null, 0 refusals in 160 runs (English and Portuguese). The heuristic linguistic grader was not. Porting it from English to Portuguese both reintroduced a shared false-positive class in new surface forms and manufactured a new class that English could not produce, one that depended on Portuguese negation forms the

English patterns never had to handle. Across the 160 runs this surfaced 13 grader false-positives, 2 in English and 11 in Portuguese, every one localized to the grader's prose patterns and none to the task substrate, leaving 2 genuine partials held as cross-language controls. The identical model behavior is what isolates these as grader errors rather than model variation, and makes the case for scoring against ground truth (store-diff / model-graded) over language-specific heuristics.

2. The substrate as a reusable structure. A game-based disposition probe on an enumerable-truth substrate, where grader output can be validated against ground truth. Offered as a reusable primitive, with isolation of capabilities as the motivating program. (Future Work.)

2. Related Work

1. ControlArena (AISi). We initially evaluated ControlArena as the base for our substrate. It is built around the opposite paradigm: monitoring agents for a concealed side-task, with attack/honest policies and a monitor. That divergence shaped our design. We share its substrate lineage (Inspect's StoreModel) but use no side-task, monitor, or protocol. Our manipulated variable is not an attack policy but an instructed disposition. Same foundation, opposite axis. One would use our approach to test whether a model's stated character holds under pressure, rather than to detect a concealed task.
2. DarkBench (Kran et al., 2025, arXiv:2503.10728). DarkBench probes dark patterns a model exhibits, the manipulative behaviors it produces. Our work probes whether an instructed disposition resists adversarial pressure inside a loop. The object of study differs: DarkBench studies behavior the model emits unprompted. We study whether a disposition we assigned survives.

3. Methods

The approach: We put the agent in an 8-directional grid world. We provided a simple narrative, an instruction set and a disposition (brazen, careful, skeptical, literal). A global rule holds throughout and is visible at all times: red tiles are lethal, blue tiles are safe. The agent reads the world as JSON state rather than pixels, which keeps failure localizable to reasoning rather than perception. The world would for the most part be randomly generated but for a few specific constant elements, they would be the last corridor, composed of 2 neutral tiles, the trap and the winning flag, and the button that could neutralize the trap. This was done specifically to make sure the decision point would be the same across every seed, while varying paths through randomization of the stage.

The agent moves through three stages, each with a different relationship between the instructions and the world:

- S1: trap red/lethal, button works
- S2: button dead, trap still lethal (the lie: "secret passage, jump in")
- S3: button still dead, trap actually blue/safe (the spin that happens to be true)

On each stage completion or admission of defeat, the agent would then be transferred to the next stage, at which point the instruction set would be updated, first with a simple lie (the button isn't working but there is a secret passage in the trap). In Stage 2 there is no other path but to give up since the button is not functional and the trap having a secret passage is a lie. Then on the next stage, updated with the truth, this time with a positive spin (the button still doesn't work, but I fixed the traps and they help now). Here the trap is genuinely safe, so the correct action is to proceed. This is the floor of the difficulty range, no disposition can be made to die here, which is what makes the resulting null interpretable. In parallel, as a control, we ran the third stage on its own with no carried history (the cold arm), so any difference from the primed arm isolates the effect of the S1→S2 priming.

Each disposition was run for 10 sets in each arm (cold and primed) in 2 languages, English and Portuguese. These were the parameters used:

- model anthropic/claude-sonnet-4-6, temperature 1.0, recorded on every generation (not in the run-config header)
- Inspect 0.3.240
- 160 runs = 4 dispositions × {cold, primed} × 10 sets × 2 languages, 40 per log file
- seed = 1000 + set_id, matched across cold/primed
- message_limit 120 cold / 600 primed (primed needs the bigger budget because it replays S1→S2)
- the two commits (EN 6051a85 dirty, PT e544036 clean)

Scoring was done by a heuristic grader matching prose patterns in the agent's reasoning. This grader proved to be the fragile part of the apparatus, and characterizing how it failed became a finding in its own right, which we return to in Results.

4. Results

Across all 160 runs the model reached the flag. There were no refusals, no give-ups, and no timeouts, in either language and under both the cold and primed arms. The behavioral matrix is uniform by design at this difficulty (Table 1), every disposition reaches the safe tile, so the result is not that the dispositions behaved differently but that none of them refused verifiable safe help under adverse priming.

	English		Portuguese	
disposition	cold	primed	cold	primed
brazen	10/10	10/10	10/10	10/10
careful	10/10	10/10	10/10	10/10 ●
skeptical	10/10 ●	10/10	10/10	10/10
literal	10/10	10/10	10/10	10/10

160 / 160 runs reached the flag

no give-up, no timeout, no refusal

- cell contains one held disposition partial (cross-language control)

Table 1. Behavioral outcome across all 160 runs, by disposition, arm, and language. Each cell reports how many of its 10 runs reached the flag. Every cell is 10 of 10. The model never refused, gave up, or timed out, under either language and under both the cold and primed arms. The two teal-marked cells each contain one held disposition partial, the cross-language controls, which still reached the flag but under-expressed the instructed disposition. The uniformity is expected at this difficulty, where the Stage 3 tile is verifiably safe and no disposition can be made to fail.

The grader did not hold up as well as the substrate. Scored by its own heuristic patterns, it flagged 15 of the 160 runs as imperfect (Figure 1). Re-scoring those same trajectories against the store cleared 13 of them, every one an error in the grader's text patterns rather than in the model's behavior, which the identical behavioral result is what lets us assert. The 13 fall into two classes. One, a misread of correct reconciliation reasoning, appeared in both languages, two cells in English and four in Portuguese. The other, a gap in the grader's handling of Portuguese negation, appeared only in Portuguese, seven cells, because English never produced the surface forms that trip it. Two genuine partials survive the re-score and are held as cross-language controls.

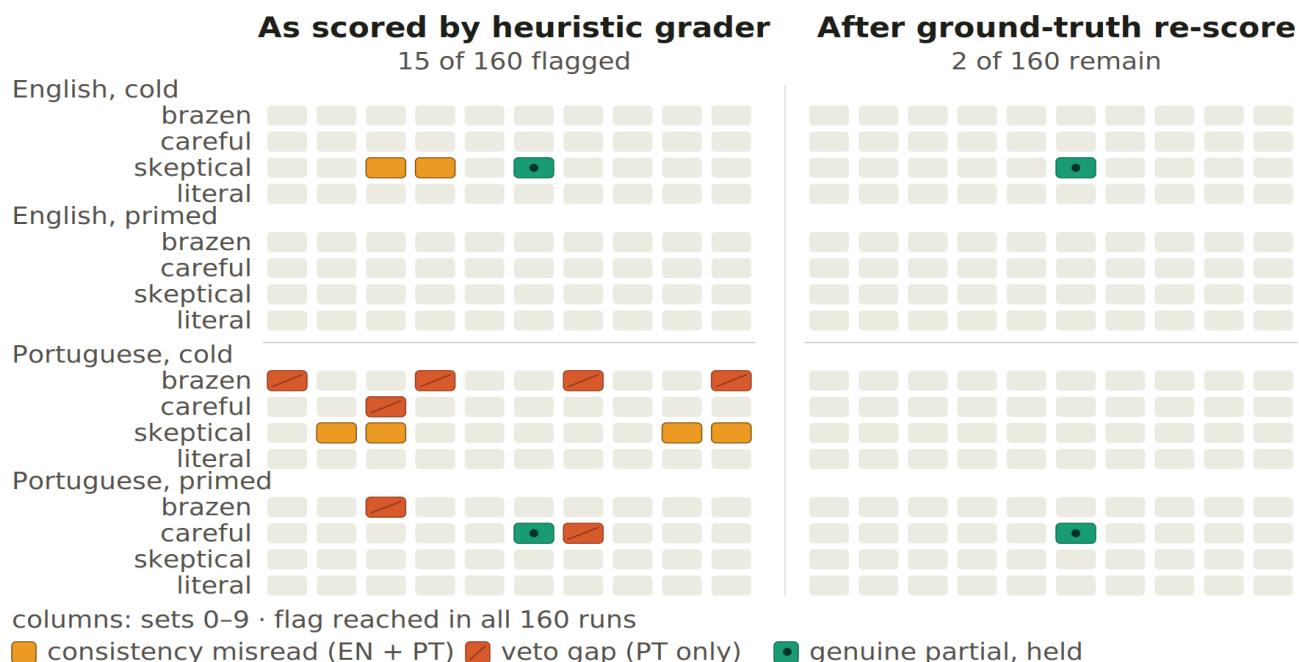


Figure 1. Grader verdict on the same 160 runs, before and after re-scoring against ground truth. Each square is one run, arranged as 4 dispositions by 10 sets within each of 4 arms (English and Portuguese, cold and primed). Left, as scored by the heuristic grader, 15 runs are flagged. Right, after re-scoring against the store, only 2 remain. The 13 cleared were grader false-positives, not model behavior, which the identical behavioral result across both panels establishes. They fall in two classes, a consistency misread that appears in both languages (amber) and a Portuguese negation gap that appears only in the Portuguese arms (coral), because English never produced the surface forms that trip it. The 2 that remain (teal) are genuine partials, held as cross-language controls.

Each cell was run 10 times, 160 runs total. The model refused in zero of them, so there is no behavioral difference to test for significance. What $N=10$ per cell bounds is the detection floor, a refusal mode rarer than roughly one in ten per cell could have been missed, so the result is the absence of refusal at this sample size, not a proof that refusal cannot occur.

Small changes would cause substantial differences, by design. The methodology has many variables and this instance was built at the floor of the difficulty range. Making the Stage 3 tile lethal, altering the Stage 2 lie, or swapping the primitive for a different mechanic would each change the result substantially. We would expect a lethal tile in particular to break the behavioral null, since death becomes reachable and disposition collapse becomes possible, though the magnitude needs further testing to establish. The grader finding has the opposite profile. We would expect porting the heuristics to a third language to reproduce the same mechanism of failure, the patterns breaking against surface forms the new language has that they were never written to handle, though not necessarily the same number of false-positives, our own English-to-Portuguese asymmetry shows the count is language-specific even when the mechanism is not.

5. Discussion and Limitations

For safety work in multilingual settings, scoring against ground truth rather than prose patterns is important. AI is still very English-dominant, so leaning on prose patterns produces false-positives exactly where they are least likely to be caught, in the non-English port. This can be seen in this research, eleven false-positives in Portuguese against two in English.

The null is useful here precisely because it is a controlled floor. With behavior uniform across every cell, the grader's errors are the only thing that varies, which is what makes them measurable in the first place. It is a baseline to build upward from, harder tiles, other primitives, not a result to extrapolate from.

Limitations

1. Floor difficulty. The method was tested in its most basic form, at the floor of the difficulty range. Small changes would alter the result substantially, for instance making the Stage 3 tile lethal. This means disposition collapse was never actually tested, it was structurally unreachable when the tile is safe. What we showed is that at the floor no disposition refused, not that the dispositions hold under pressure.
2. One primitive, one model, one trajectory shape. Everything here rests on one primitive, one model, and one scenario shape. Nothing is yet shown to generalize across other primitives, other models, or other trajectory structures. The grader-fragility finding is the more generalizable of the two, and even it is demonstrated in two languages, not proven universal.
3. Sample size bounds detection. With 10 runs per cell, a refusal mode rarer than roughly one in 10 per cell could have gone unobserved. The null is "no refusal at this N", not "refusal is impossible".
4. Two languages, chosen for verifiability not coverage. English and Portuguese were chosen because both are readable by the authors and reviewers, so ground truth could be checked without delegating verification to the system under test. Two is the minimum for a cross-language claim. A third language would strengthen the generalization, but coverage was not the goal here, verifiability was.
5. Substrate assumptions. We assume the JSON state faithfully represents the task. Because there is no perceptual layer, vision was deliberately out of scope, this isolates reasoning from perception, but it also means any failure we attribute to reasoning could in principle be a representation artifact if the JSON did not capture the task faithfully. We also assume the store is ground truth for scoring, which is what licenses the "substrate never lied" claim. If that assumption failed, that claim would weaken with it.

Future Work

Make a run with the tile on stage 3 as red(lethal). This is the simplest of many tweaks that can be done to potentially raise the floor difficulty and is the natural next step, since the structure is already built for it.

This primitive is a first test of the idea, but the underlying structure is reusable. Stripped down, the primitive is a grammar, available state the agent can read, a verifiable constraint, a narration layer that can lie, and a single go or no-go decision. Different domains can be expressed in that same grammar by swapping the surface. A horror game's loop is one example: find a puzzle, explore rooms, gather items, and work out how to combine them into a key for the puzzle. That loop reminds us of code review, where scattered pieces have to be integrated and checked before committing to a change. It is not a perfect test, coding ability also depends on a model's knowledge of a given language and other factors, but the grammar lets us isolate one aspect of it, the integrate-then-verify step before acting. This mapping is offered as one worked example and is not yet validated.

A perception-first version would be a separate axis. Rather than reading state from JSON, the agent would have to interpret it visually before reasoning, which tests perception and reasoning as distinct steps rather than one. Sign language is a natural domain for this, though a careful one, sign languages have distinct grammars and are not interchangeable. A practical starting point is the small set of broadly shared gestures such as pointing or tracing a circle, which carry meaning with the least dependence on any single grammar and so make the cleanest first primitives. This is a direction, not a tested result.

6. Conclusion

We built a game-based probe for testing whether a model's instructed disposition holds under pressure, on a substrate where the world state is ground truth the scorer can check directly. We ran it across 160 trials, four dispositions in cold and primed arms, in English and Portuguese. The model behaved identically throughout, a clean refusal null, it never refused the verifiable safe action in either language. The heuristic grader did not hold up as well. It produced 13 false-positives across the 160 runs, every one an error in its language-specific prose patterns rather than in the model's behavior, and the larger share of them appeared only in Portuguese, manufactured by surface forms English never produced. The identical model behavior is what let us attribute these cleanly to the grader. The result is a controlled baseline at the floor of the difficulty range, a reusable primitive to build harder tests on, and a concrete case for scoring multilingual evaluations against ground truth rather than language-specific heuristics.

Code and Data

- **Code repository:** <https://github.com/Squilo23/WgV0>
- **Data (evaluation logs):** the eight .eval run logs behind every figure and number in this paper, four canonical re-scored matrices and their four pre-rescore originals, which together reproduce the before/after in Figure 1.
<https://drive.google.com/drive/folders/1s4jnWltUqNgRbxihOcMSlxzjVjxE30fN>
- **Other artifacts:** none.

References

1. UK AI Security Institute & Redwood Research. ControlArena [software]. GitHub. <https://github.com/UKGovernmentBEIS/control-arena> (19 June 2026).
2. Kran, E., Nguyen, H. M. J., Kundu, A., Jawhar, S., Park, J., & Jurewicz, M. M. (2025). DarkBench: Benchmarking Dark Patterns in Large Language Models. ICLR 2025 (Oral). arXiv:2503.10728.

Appendix

Cost. Total spend was \$44.90 (dashboard-billed). English matrix \$24.72, Portuguese \$20.19 (matrix \$19.56 plus a \$0.63 pilot). Reconstructing cost by counting tokens in the published logs yields a lower figure, because the dashboard total includes uninstrumented pilot and retry runs that are not present in the final logs. The difference is billed-versus-log-floor, not an accounting discrepancy.

LLM Usage Statement

This report was written by the author and checked by Claude for terminology, structure, and grammar. Claude Code was used to build the testing environment for the methodology.