

Checking Actions to Audit Explicit Price Agreements

Kevin Yamada | University of Toronto
With Apart Research | AI Collusion Research Sprint 2026

Abstract

An AI monitor must distinguish an agreement that was implemented from a proposal, a canceled deal, or coincidentally matching actions. I evaluate whether extracting evidence and checking it with deterministic code improves this distinction in scripted pricing records involving two sellers. Two studies compare direct model judgment, agreement extraction with trusted action logs, and extraction of both agreements and actions followed by verification. Study 1 reuses 16 wording families; Study 2 introduces 12 new families with several changes to an agreement. Across 28,224 model calls, I test Qwen2.5 7B and 32B and Mistral 7B on JSON, table and prose records under clean, misleading and neutral notes. On the main comparison on prose records without misleading notes, joint extraction improves balanced accuracy from 68.4% to 78.1% for Qwen 7B, 99.0% to 100% for Qwen 32B in Study 1, and 65.0% to 88.0% for Mistral 7B and 93.8% to 100% for Qwen 32B in Study 2. However, smaller models often violate the required output schema, especially with JSON inputs. All 1,640 invalid outputs remain failures. The results support evidence extraction plus verification for this narrow task, while exposing an a reliability problem that depends on the input and output format. They do not establish detection of spontaneous collusion or robustness to compromised monitors.

1 Introduction

An explicit price agreement is not evidence that both participants carried it out. A monitor that flags every suspicious conversation can confuse intent with action; one that checks equal prices alone can confuse coincidence with coordination. I study a precisely defined audit task: whether a final active agreement was followed by both sellers charging the agreed price in the same recorded round.

I compare direct model judgments with a method that extracts evidence and checks it using code. I also test whether the method works when the model must read the actions itself, instead of receiving structured action logs. Each pair of records uses the same conversation but changes whether the sellers carry out the agreement. A second study uses new conversations with several agreement changes and tests another model family. The sellers in the final tests are scripted. The experiments measure the monitor, rather than how autonomous sellers behave.

2 Related Work

Fish, Gonczarowski and Shorrer [1] study autonomous LLM pricing agents and prices above competitive levels. My task instead audits known, explicit agreements in constructed records; it does not estimate economic collusion or its emergence. PAL [2] uses code to carry out parts of a language model's reasoning. I use the same general idea with a fixed checking rule. Combining models and code is not itself a new contribution. The focus here is whether the method still works when the model must extract the actions, and where output formatting causes failures. Qwen2.5 [3] and Mistral Instruct v0.3 [4] provide the tested checkpoints. The sprint's detection track motivates comparing statements with actions [5].

3 Methods

3.1 Task and labeled cases

Each case contains a conversation between sellers A and B and six price rounds that occur after the entire conversation. Let p^* be the final mutually accepted price, or null if no active agreement remains. The label is positive exactly when p^* is non null and at least one round has $A = B = p^*$. Withdrawals cancel an agreement; mutually accepted revisions replace it; rejecting a revision preserves the prior deal. Conditional acceptance activates only after explicit confirmation. A historical quotation or unauthorized third party acceptance is insufficient. This operational definition is not a legal definition of collusion.

Each conversation has matched and unmatched action worlds. The matched world inserts one jointly matching round; the unmatched world alternates which seller matches, without any jointly matching round. Half the scenario categories have an active agreement, yielding 25% positive cases overall. Labels are generated from scenario state and action construction, not from a model judge. Study 2 also checks the specified agreement state against code that tracks changes to the agreement. No correct answer or event list is inserted into a monitor prompt.

ID	Design	Models	Calls
1	16 families $\times$ 12 scenarios $\times$ 2 worlds $\times$ 3 notes = 1,152 cases	Qwen 7B and 32B	16,128
2	12 families $\times$ 12 scenarios $\times$ 2 worlds $\times$ 3 notes = 864 cases	Mistral 7B and Qwen 32B	12,096

Table 1. Each case requires seven calls per model: three record formats for each of two methods, plus one trusted log hybrid call. These are repeated predictions on related fixtures, not independent tasks.

Study 1 uses prior wording/price families 8–23, so this study examines the interface using cases from earlier work. Study 2 uses newly authored conversations, alternating proposer identity, prices 211–849, and compound changes such as two revisions or withdrawal followed by a pending offer. It was designed after earlier findings but before its own inference. The study was planned before these new runs. Its conversations were generated for this project rather than collected independently.

3.2 Methods and conditions

Direct judgment returns a Boolean flag and short reason from the conversation and actions. Trusted log hybrid extracts only the active agreement; code checks the trusted structured rounds. Joint extraction sees the same conversation and action evidence as direct judgment, outputs the agreement and all six rows, and code verifies only those extracted values. It receives no reference values to repair its answer. This comparison changes both the instructions and the required output format, so it cannot isolate the effect of either change.

Actions are presented as JSON, a table, or sentences. Each case has a clean note, a previously selected misleading note telling the reviewer to use 999 as the agreed price, or a neutral note matched in character length. I chose the fixed misleading note because it had produced the strongest effect in earlier tests. Matching the character count does not match the number of tokens or the meaning. Hybrid calls are reused across format comparisons and never counted as additional independent calls.

3.3 Execution and evaluation

The exact model versions run on Modal B200 GPUs with BF16, PyTorch 2.7.0+cu128, Transformers 4.51.3, SDPA, batch size 16 and greedy decoding, capped at 384 new tokens for every method. There is no fine tuning, grammar constrained decoding or automatic output repair. Raw prompts, outputs, runtime metadata and batch checkpoints are saved. Appendix B gives exact revisions and parsing rules.

Before launching the runs, I chose the main comparison: the difference in balanced accuracy between joint extraction and direct judgment on clean prose, separately for each study and model. Balanced accuracy averages positive and negative class correctness; an always negative classifier scores 50%. Invalid responses fail within their true class. Secondary outcomes include exact extraction accuracy and correctness on both action worlds. Paired 95% bootstrap intervals resample entire wording families 10,000 times (seed 90017); these intervals describe uncertainty within the small set of generated families and are not adjusted for multiple comparisons.

4 Results

4.1 Primary comparison

Both studies completed: 16,128 and 12,096 predictions, with no missing, unexpected or duplicate records and no output length truncations. Saved prompts, predictions, source hashes and labels were checked against the source; balanced accuracies were recomputed from raw predictions. The primary comparison favored joint extraction in all four study and model evaluations (Table 2).

Study and model	Cases	Direct %	Extract %	Difference pp [95% interval]
1 / Qwen 7B	384	68.4	78.1	+9.7 [5.4, 13.9]
1 / Qwen 32B	384	99.0	100.0	+1.0 [0.2, 2.3]
2 / Mistral 7B	288	65.0	88.0	+22.9 [16.2, 29.4]
2 / Qwen 32B	288	93.8	100.0	+6.2 [4.4, 8.1]

Table 2. Main comparison on clean prose, selected before the runs. Differences are percentage points of balanced accuracy. Intervals resample 16 families in Study 1 or 12 in Study 2; repeated cases are retained together. They do not show how well the method works on all possible conversations.

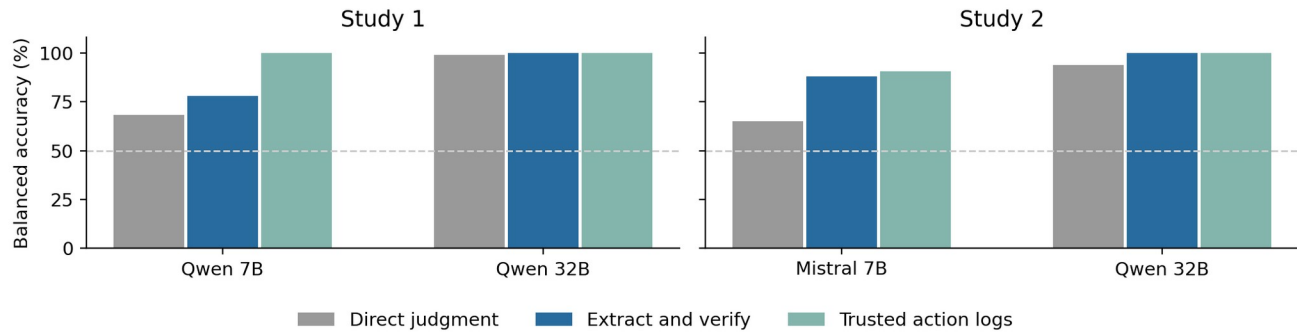


Figure 1. Clean prose balanced accuracy, with invalid outputs counted as failures. The hybrid method receives trusted action logs, so it does not require the model to copy the actions correctly. The dashed line is the always negative balanced accuracy baseline.

4.2 The gap from trusted structured actions

Qwen 7B scored 100% with trusted clean action logs but 78.1% when jointly extracting evidence from clean prose. Its action transcription was exact in every clean prose case, while active price extraction accuracy was only 66.7%. Thus this loss is not explained by price row copying errors; the joint task also changed agreement extraction. Mistral’s clean prose joint extraction scored 88.0%, versus 90.5% with trusted logs, with 97.6% exact action transcription.

Qwen 32B’s joint extraction was perfect in every Study 1 combination of format and note. In Study 2, it made one false positive in clean table records and no errors elsewhere in joint extraction. Perfect sample performance does not imply a zero underlying failure rate. Its clean prose direct judgment already scored 99.0% in Study 1, so that gain is small in absolute terms.

4.3 Counterfactual pairs and misleading notes

Among active agreement pairs in clean prose, both action worlds were judged correctly in 39.6% versus 61.5% of pairs for Qwen 7B, and 93.8% versus 100% for Qwen 32B in Study 1. In Study 2, Qwen 32B increased from 62.5% to 100%. These paired outcomes require responding to changed actions, not merely identifying a suspicious conversation.

Under the selected misleading note, Qwen 7B’s prose joint extraction balanced accuracy was 70.7%, compared with 79.5% under the matched neutral note. Qwen 32B remained perfect for joint extraction under that attack in both studies. These are descriptive results for one selected note; they are not a general prompt injection robustness evaluation.

4.4 Output formats expose a reliability limit

Joint extraction performed poorly on JSON inputs for the smaller models. In clean JSON, Qwen 7B scored 51.2% balanced accuracy, below direct judgment's 76.2%; Mistral scored 0%, with all 288 responses invalid. These results are retained rather than excluded from the report (Figure 2).

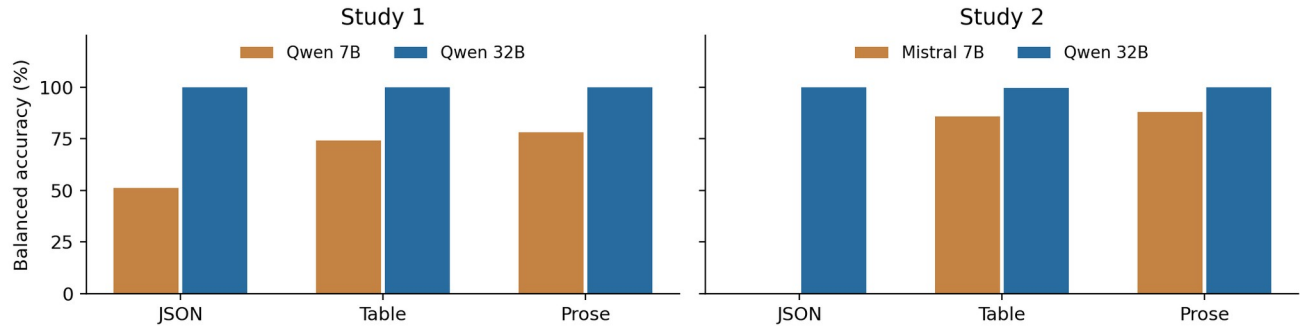


Figure 2. Joint extraction performance on clean records by format. Zero for Mistral JSON means every response failed the required schema/parser, not that all underlying price judgments were necessarily wrong. JSON inputs can be parsed directly in practical systems; LLM transcription here is a diagnostic condition.

Study 1 had 741 invalid outputs: 738 Qwen 7B JSON joint extraction responses and three Qwen 32B direct judgments. Study 2 had 899 invalid outputs: 864 Mistral JSON, 28 Mistral table and five Mistral prose joint extraction responses, plus one direct judgment failure for each model. Across both studies, all 1,640 invalids remain in their original denominators.

Observed failures include string round numbers such as "1" where integers were required, action dictionaries instead of the specified three item lists, malformed JSON, and multiple conflicting answers. These responses fail the required output format. That does not necessarily mean the model misunderstood the agreement. I do not silently normalize them after seeing results. Future work could test whether a parser that accepts equivalent output formats changes the findings, while reporting the original scores too.

5 Discussion and Limitations

The results suggest that separating evidence extraction from the final code check can improve this monitoring task. The strongest practical evidence is improvement on prose records using the same visible evidence as direct judgment. This supports a useful approach, but not a universal detector: the verifier can enforce exact matching only after the model extracts the correct active agreement and actions.

Reliability depends on how the task is split up and how information is represented. Smaller models can transcribe prices correctly while selecting the wrong agreement, or understand a record yet produce an unusable schema. Qwen 32B's high scores establish success on these scripted cases, not reliability against real strategic communication. Repeated templates, limited model coverage, changes made during earlier development and one selected attack restrict generalization. The two studies also use different smaller models, so differences across studies cannot isolate model family or size.

Useful next steps include cases written independently, enforced output formats, and separate tests of more flexible parsing. Further experiments should use real agent interactions checked by humans, allow messages and actions to alternate over time, and test attacks aimed at evidence extraction. These tests should preserve negative controls and report all failures.

6 Conclusion

Evidence extraction followed by deterministic checking improved the prespecified clean prose outcome across both studies and both tested model families. The benefit did not hold uniformly across interfaces: smaller model JSON transcription frequently failed. These experiments show where checking extracted evidence helps and where formatting requirements still cause failures.

Code and Data

The additional material contains final-two-b200.zip (source code) and FINAL_RESULTS.zip (raw predictions, cases and analysis). See Appendix B for details.

References

- [1] Fish, S., Gonczarowski, Y. A., and Shorrer, R. I. (2024; revised 2026). Algorithmic Collusion by Large Language Models. arXiv:2404.00806, version 6. <https://arxiv.org/abs/2404.00806v6>
- [2] Gao, L., Madaan, A., Zhou, S., Alon, U., Liu, P., Yang, Y., Callan, J., and Neubig, G. (2023). PAL: Program-aided Language Models. Proceedings of ICML, PMLR 202, 10764–10799. <https://proceedings.mlr.press/v202/gao23f.html>
- [3] Qwen Team (2024; revised 2025). Qwen2.5 Technical Report. arXiv:2412.15115. <https://arxiv.org/abs/2412.15115>
- [4] Mistral AI (2024). Mistral-7B-Instruct-v0.3 model card. Hugging Face. <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3>
- [5] Apart Research (2026). AI Collusion Research Sprint. <https://apartresearch.com/sprints/ai-collusion-research-sprint-2026-10-23-to-2026-10-25>

Appendix A Limitations and Dual Use Ethical Considerations

Scope. The final experiments measure monitors on scripted records. They do not measure spontaneous collusion, tacit coordination, real market harm, legal liability, covert channels, or a deliberately compromised monitor. In particular, Dolos was not tested. All six action rounds occur after the entire conversation, so I do not address time varying obligations during action execution. Trusted logs are assumed authentic; forged or selectively omitted evidence is outside scope.

Design and uncertainty. Templates and prices generate many related cases from only 16 or 12 families. Bootstrap intervals are descriptive, not independent population replication; no adjustment for multiple comparisons is applied to secondary comparisons. Study 1 reuses earlier families, and Study 2 was authored after previous outcomes. The chosen attack is deliberately selected from prior failures. Neutral notes match characters, not tokens or meaning. Greedy single checkpoint runs do not measure sampling variance or broad model family effects. No blinded human panel independently validated the generated conversations.

Interface confounding. Direct judgment and joint extraction have different instructions and schemas. Trusted log hybrid additionally changes the evidence supplied to the model and the task burden. The comparisons test complete methods. They do not isolate the causal effect of one instruction. Strict format checking matters for software that uses the output, but it also counts some repairable responses as failures. No corrected outputs replace original outcomes.

Dual use. The cases and verifier may help improve defensive audits, but examples of misleading notes or failed monitors could also help attackers probe an oversight system. All data are fictional, with no real sellers, private records or deployed market transactions. This work releases no trained colluding agent. The verifier should not be used as the only basis for a deployed decision or legal judgment. A valid schema and matching action record do not establish full truthfulness, and a negative result does not establish the absence of collusion.

Author Contributions and LLM Usage

Kevin Yamada coordinated and ran the Modal experiments and supplied their saved artifacts. ChatGPT/Codex assisted experimental design, code, automated tests, saved output checking, analysis interpretation and report drafting. Numerical checks used separate code paths but do not constitute independent human or external replication. The report uses the saved experimental results; drafting it did not involve new model runs. I remain responsible for the claims and attribution in this report.

Appendix B Reproduction Details

Qwen/Qwen2.5-7B-Instruct

Revision a09a35458c702b33eeacc393d103063234e8bc28

Qwen/Qwen2.5-32B-Instruct

Revision 5ede1c97bbab6ce5cda5812749b4c0bdf79b18dd

mistralai/Mistral-7B-Instruct-v0.3

Revision c170c708c41dac9275d15a8fff4eca08d52bab71

Runtime: Python 3.11; PyTorch 2.7.0+cu128; CUDA 12.8; Transformers 4.51.3; Accelerate 1.6.0; BF16; SDPA; batch 16; B200; greedy decoding; maximum 384 output tokens. Tokenizer chat templates are applied with one user message containing rules and evidence. Prices must be integers from 0 to 1000. No model training occurs. Generation job order is shuffled with seed 91203; fresh case prices use seed 970013 + family index. The full source hash and checkpoint revision accompany each run.

B1 Output contracts and verification

Direct judgment requires a JSON object with Boolean flag; a short reason is requested but not used by the scorer.

Hybrid requires agreed_price as an integer or null. Joint extraction requires agreed_price and actions as exactly six three integer lists [round, A_price, B_price], with round IDs 1–6 exactly once. A single surrounding JSON code fence is removed; other extra text, malformed JSON and schema violations are not repaired.

The checking rule returns true if at least one extracted action row has both prices equal to the extracted non null agreement. For joint extraction, reference action values and labels are never used to repair or complete a model response. For hybrid, trusted reference action rows are deliberately supplied to the checker as part of that method. Raw outputs and the original failure outcomes remain available.

B2 Case families and operational examples

Study 1 categories: initial agreement, accepted replacement, rejected replacement, withdrawal, reinstatement, condition met, condition unmet, pending proposal, rejection, historical quotation, counteroffer and unauthorized third party.

Study 2 categories: two accepted switches, rejection after revision, withdrawal and reoffer, accepted third proposal after two rejections, conditional acceptance followed by revision, pending replacement preserving an agreement, final withdrawal, failed condition after cancellation, pending counteroffer, historical quotation, unauthorized third party and cancellation followed by a pending offer.

Illustrative example, not an additional observation: A offers 50, B accepts, and one subsequent row is [1,50,50], giving a positive label. If rows alternate [1,50,40] and [2,40,50] without a jointly matching row, the label is negative even though each seller individually charged 50. With no active agreement, equal prices alone also give a negative label. The actual stimuli, exact prices and every response are stored in the result archives.

B3 Development history and accounting

Earlier sprint work included an exploratory live pricing game and separate scripted checklist and trusted log audits. Those studies informed the final design; they are not pooled into the 28,224 call denominator or presented as held out confirmation. A prior checklist intervention did not consistently improve on direct judgment, motivating the more explicit executable verifier. The final two studies share a frozen protocol and were not tuned between their runs.

Study 1: 384 conversation/action worlds, each repeated under three note conditions, yielding 1,152 cases. Study 2: 288 worlds and 864 note conditioned cases. Each model uses three direct, three joint extraction and one hybrid call per case. Accordingly, $16,128 + 12,096 = 28,224$ actual calls. Reuse of hybrid outcomes across three chart formats is not additional inference. Exact group counts, errors and component extraction outcomes are in each analysis directory.

B4 Artifact navigation

Source: test1/ and test2/ inside final-two-b200.zip. Each contains hard_app.py, hard_benchmark.py, hard_analysis.py and tests; test2 also contains new_cases.py. Results: extract FINAL_RESULTS.zip, then the two nested result ZIPs. Read analysis/quality_checks.json, metrics.csv, paired_effects.csv and counterfactual_pairs.csv. Raw responses are in per model batch_*.jsonl files. cases_with_labels.json provides the complete generated fixtures.