# LEARNING TO ELICIT: CO-EVOLVING ROUTING AND INFORMATION DISCLOSURE IN SOCIETIES OF LANGUAGE AGENTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Multi-agent language models increasingly operate in societies of agents characterized by information asymmetry, where the information required to solve a task is distributed across agents. Effective coordination therefore depends not only on solving the task but also on deciding what information to elicit and from whom. We study this problem in a multi-agent meeting environment where agents possess private facts and must communicate over a shared public record to reach a verifiable settlement. We introduce ELICITBENCH, a benchmark in which agents hold distributed private information and must reveal the relevant facts through interaction before a designated decision maker can reach the correct outcome. We then propose CO-ADAPT, a co-evolutionary training framework that jointly learns routing and agent information disclosure through reinforcement learning and privileged on-policy distillation. The router learns which agents to engage, while agents learn which private information to reveal. On the held-out test set, Co-Adapt raises the fraction of settlement checks passed from 0.65 to 0.81 and the strict success rate from 14% to 25%. Gains are largest on domains unseen during training, where the fraction of settlement checks passed increases from 0.57 to 0.78. Ablation studies further demonstrate that the gains arise from jointly learning information routing and information disclosure, rather than from either component alone.

## 1 INTRODUCTION

Knowledge in a group is rarely distributed evenly: the facts needed for a decision may be held by different members, while the person responsible for deciding sees only a fraction of them. This creates a well-studied *hidden profile*: groups tend to discuss information that members already share, while information known only to one member is less likely to be surfaced, even when it is critical to the correct decision (Stasser & Titus, 1985; 2003; Lu et al., 2012). This failure is not simply a matter of incomplete communication. Members often have no reason to know which pieces of information are missing, and therefore discussions naturally sample from what is already salient or mutually known (Stasser & Titus, 1987; Larson et al., 1994). As language models move from isolated assistants to societies of interacting agents, they face the same information asymmetry. Each agent may possess a different private view of the task, making successful coordination depend not only on reasoning about the task, but also on discovering *who knows what* and eliciting the relevant information. Recent evaluations suggest that LLM societies can reproduce these collective reasoning failures: when information is distributed across agents, groups can perform substantially worse than a single agent given the complete information (Li et al., 2025). Thus, giving multiple agents access to one another does not by itself ensure that a society will pool the information it needs.

Yet existing work leaves an important question open: *how should agents learn to elicit information that they cannot observe?* Most multi-agent evaluations either provide agents with a shared or omniscient view of the task, or study interaction without requiring recovery of a hidden decisive fact (Zhou et al., 2024a). Other benchmarks study information asymmetry, inquiry, disclosure, or theory of mind, but often rely on prompting-based interventions or model-based judgments of interaction quality (Zhou et al., 2024b; Kim et al., 2023; YS et al., 2026; Li et al., 2025). Routing and orchestration methods similarly learn which agents are useful from observable signals of compe-



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tence (Liu et al., 2023). In a hidden-profile setting, however, the decisive property: *whether an agent holds a missing fact is itself unobserved.* Human studies suggest that awareness of who knows what can improve information pooling (Stasser et al., 1995; 2000; Larson Jr et al., 1998), but successful coordination alone does not establish that an agent has learned to recover otherwise inaccessible information. Without verifying that the same task is difficult or unsolvable from an individual agent's available information, improved group performance may instead reflect the additional capability obtained from multiple model instances. These gaps make it difficult to isolate *information elicitation* as a distinct capability of multi-agent systems. Figure 1 illustrates a representative hidden-profile meeting in which critical information is distributed across participants.

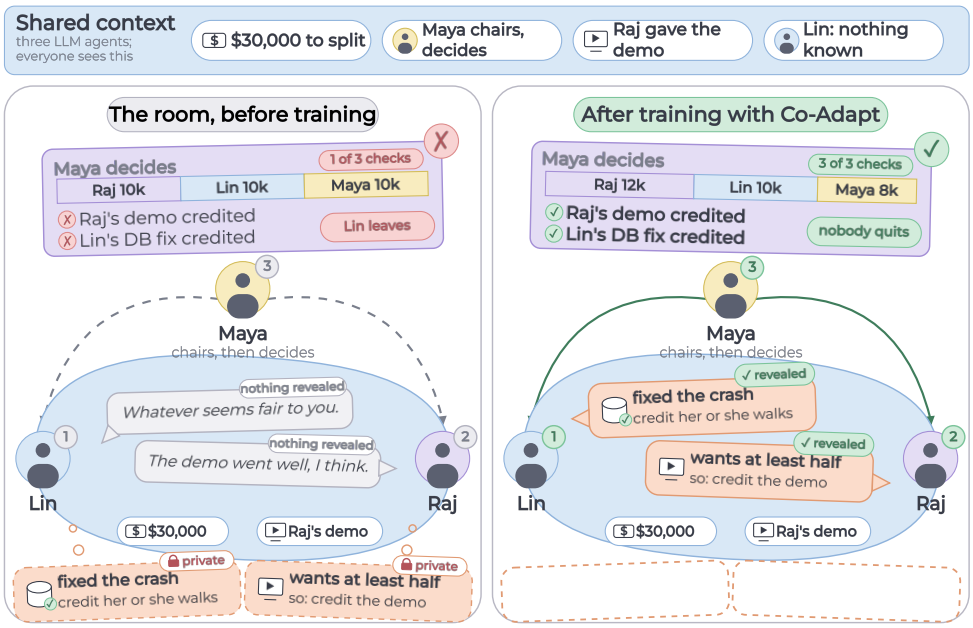



Figure 1: Three agents must split a $30,000 bonus. Lin secretly fixed a crash that saved the launch and has another job offer, while Raj gave the demo and wants at least half the bonus. Maya leads the meeting and decides how to split the money. Before CO-ADAPT, Lin and Raj keep their private information hidden, so Maya splits the money evenly and only one check passes. After CO-ADAPT, they reveal their private information, Maya recognizes both contributions and adjusts the split. As a result, all three checks pass and nobody quits.

To study this problem, we introduce ELICITBENCH, a benchmark of 1,100 meeting scenarios spanning 11 domains. Each scenario contains 3–5 participants, holding private facts, together with a designated decision maker and a verifiable settlement. The scenarios are constructed as hidden profiles: decisive facts are distributed across participants such that recovering those facts is necessary for the final settlement to pass. We construct the benchmark through a process involving both humans and language models: human-written seeds specify the intended decision, participant roles, and hidden information structure; model-generated scenarios are then iteratively verified and revised, with human editors checking cases for realism and plausible information withholding. We evaluate both the content of the settlement and the provenance of the information used to construct it through programmatic checks, avoiding an LLM judge. In particular, a private fact counts as elicited only when its owner formally reveals it into the shared record (Clark & Brennan, 1991); merely mentioning a fact in free-form conversation is insufficient. This distinction allows us to separate genuine information acquisition from producing a correct answer for the wrong reason. We further certify each scenario by requiring a lone agent to fail while a coordinated group can pass under the same verifier, and hold out two domains entirely for evaluating whether learned elicitation strategies transfer beyond the training distribution.

We then introduce CO-ADAPT, a co-evolutionary framework that jointly learns *routing* and *reasoning*. A chair agent maintains the shared interaction state (Wegner, 1987) and chooses which participant to engage next, while participants reason about their private information and decide what

to reveal. This division reflects two distinct challenges in information-poor societies: the chair must determine *where to look*, while participants must determine *what to share*. The distinction is also important for learning. A privileged teacher that observes all private facts can readily identify which information is missing from the public record, but it cannot directly demonstrate whom a partially informed chair should query: once the teacher already knows every fact, the information-seeking problem has disappeared (Zhou et al., 2024a). We therefore use the privileged view to distill descriptions of the information state (Agarwal et al., 2024) while learning routing from interaction with the verifier via reinforcement learning (Shao et al., 2024).The goal is to equip the student agent with better signals for inferring otherwise private information at inference time, providing a stronger learning signal that can accelerate convergence and improve the robustness of the resulting social agency. The two policies are trained alternately so that they co-adapt: improved routing exposes more useful information, while improved participant disclosure makes subsequent queries more informative.

In summary, our contributions are:

1. **Social simulation formulation:** We introduce a verifiable multi-agent social simulation for information elicitation under distributed private information and adaptive interaction.
2. **ELICITBENCH**: A certified hidden-profile benchmark for studying information elicitation, with programmatic verification of both settlement correctness and information provenance.
3. **CO-ADAPT:** A co-evolutionary training framework that jointly optimizes information-aware routing and participant disclosure through verifier-driven reinforcement learning and privileged on-policy distillation, with alternating updates between the chair and participant policies.
4. **Empirical evaluation:** We show that co-evolving routing and participant disclosure improves information pooling and generalizes to domains unseen during training.

## 2 RELATED WORK

**Information asymmetry and agent routing.** Societies of LLM agents have been studied for collective outcomes such as sustaining a shared resource (Piatti et al., 2024). A growing line of work studies language agents that interact under private information, goals, or beliefs. SOTOPIA (Zhou et al., 2024b) and its follow-up study on information-asymmetric simulation (Zhou et al., 2024a) examine social interaction when agents cannot access one another's private state, while FANToM (Kim et al., 2023) evaluates whether models can track what different participants know, and ConfAIde (Mireshghallah et al., 2024) whether they respect who should be told what. Multi-agent benchmarks such as iAgents/InformativeBench (Liu et al., 2024) and SOTOPIA-TOM (YS et al., 2026) extend this setting to information exchange among several agents. Table 1 compares ELICITBENCH with existing benchmarks across the properties required for certified information elicitation. More recently, Peng et al. (2025) study training agents to ask and share information in asymmetric-information tasks, while HiddenBench (Li et al., 2025) shows that multi-agent LLMs can fail to recover distributed information even when a fully informed single agent succeeds. Separately, multi-agent routing and orchestration methods learn which agent to invoke using observable task or interaction signals, including DyLAN (Liu et al., 2023), GPTSwarm (Zhuge et al., 2024), AFlow (Zhang et al., 2025a), MasRouter (Yue et al., 2025), Router-R1 (Zhang et al., 2026b), and subsequent learned orchestration approaches (Dang et al., 2026; Ke et al., 2026; Jiang et al., 2026). In contrast, our setting requires the router to discover

| Benchmark | Hidden profile | Decision authority | Verifiable outcome | Provenance |
|---|---|---|---|---|
| SOTOPIA | ✗ | ✗ | ✗ | ✗ |
| FANToM | ✗ | ✗ | ✓ | ✗ |
| InformativeBench | ✓ | ✗ | ✓ | ✗ |
| Einstein Puzzles | ✓ | ✗ | ✓ | ✗ |
| MAGPIE | ✓ | ✗ | ✗ | ✗ |
| SOTOPIA-ToM | ✗ | ✗ | ✗ | ✓ |
| HiddenBench | ✓ | ✗ | ✓ | ✗ |
| ELICITBENCH | ✓ | ✓ | ✓ | ✓ |

Table 1: **Hidden profile**: decisive facts are split across participants. **Decision authority**: one participant settles the outcome. **Verifiable outcome**: scored against ground truth, not an LLM judge. **Provenance**: checks that required facts were disclosed.

which agent holds an unrevealed decisive fact, while simultaneously training that agent to reveal the relevant information.

**Co-evolution and privileged learning.** Recent work explores co-training multiple LLM roles or co-evolving agents with their environments. MALT (Motwani et al., 2024), MAPoRL (Park et al., 2025), OPTIMA (Chen et al., 2025a), CORY (Ma et al., 2024), Stronger MAS (Zhao et al., 2026c), CoMAS (Xue et al., 2026), and Multi-Agent Evolve (Chen et al., 2025b) jointly adapt multiple roles or agents through interaction. Other approaches co-evolve agents with environments, curricula, or skill representations, including SEAL (Hu et al., 2026), GenEnv (Guo et al., 2025b), Coevolve (Yang et al., 2026), R-Zero (Huang et al., 2026), Absolute Zero (Zhao et al., 2026a), EvoSkill (Alzubi et al., 2026), EvoSkills (Zhang et al., 2026a), and COSPLAY (Wu et al., 2026b). These approaches demonstrate the value of jointly adapting interacting components, but the components they adapt are peer roles on a shared task, an environment interface, or a skill library. A related line uses privileged self-distillation for language models and agents: the same model, given privileged context, supervises an unconditioned policy, typically on its own rollouts (Penaloza et al., 2026; Hübotter et al., 2026; Zhao et al., 2026b; Lu et al., 2026; Wu et al., 2026a), or provides turn-level credit for reinforcement learning (Wang et al., 2026b), where the privileged context is a reference solution, environment feedback, or a retrieved or hindsight skill.

## 3 ELICITBENCH: A CERTIFIED HIDDEN-PROFILE ENVIRONMENT

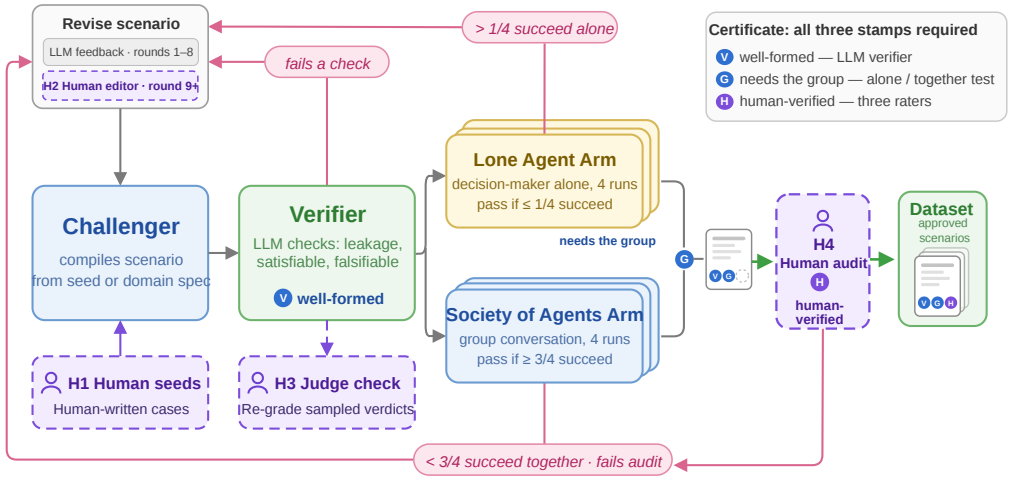



Figure 2: ELICITBENCH construction and certification pipeline.

An environment for studying information elicitation should capture four properties. (1) *Hidden profile*: where decision-critical facts are distributed across participants and absent from the decision maker's initial observation (Stasser & Titus, 1985; 2003). (2) *Verifiable*: where decisions and their information provenance admit deterministic evaluation, rather than judged by an LLM. (3) *Certified*: where the instance is verified to be unsolvable under the decision maker's initial information but solvable under interaction (Li et al., 2025). (4) *Procedural*: where instances and validity checks are automatically constructed and certified. ELICITBENCH instantiates these properties in a benchmark of meeting scenarios with distributed private information and programmatically verified decisions.

### 3.1 DATASET CONSTRUCTION

**Human Seeds.** We construct ELICITBENCH through a human–model collaborative pipeline. We first collect human-written seed scenarios describing real decisions in which the correct outcome depends on information held by different participants. Each seed specifies the decision, participant roles, relevant private facts, why those facts may not be volunteered, and how they affect the decision. Thus, the hidden-information structure originates from human judgment rather than being invented entirely by the language model. Please refer to Appendix B.4 for more details.

**Challenger.** Given a human seed, a domain specification, and recently curated scenarios from the same domain, the *Challenger* language model instantiates a complete benchmark scenario. It specifies the shared context, private facts, interaction protocol, settlement conditions, and executable checks, while avoiding information leakage and near-duplicate scenarios.

**Certification.** Each candidate is first checked by the *Verifier* for well-formedness, leakage, satisfiability, and falsifiability. We then test whether the scenario actually requires information elicitation. In the *Lone Agent Arm*, the decision maker receives only its initial view and must decide without interaction. In the *Society of Agents Arm*, all participants interact under the benchmark protocol and can reveal their private facts. A scenario is certified only when the lone agent generally fails while the interacting society reliably succeeds, ensuring that the task difficulty arises from distributed information.

**Revision and Human Review.** Failed candidates are iteratively revised using diagnostic feedback and regenerated by the Challenger. When automated refinement does not resolve a failure, a human editor examines the scenario and rollout transcripts, identifies the underlying problem, and provides a correction or discards the scenario. Human review also calibrates the automated verifier and audits released scenarios for realism, plausible information withholding, and faithful encoding of the intended private facts. Human involvement therefore enters both at the beginning, through scenario seeds, and throughout validation and refinement, rather than serving only as a final annotation step. The complete pipeline for construction of ELICITBENCH is shown in Figure 2 and the dataset statistics are shown in Appendix B.6.

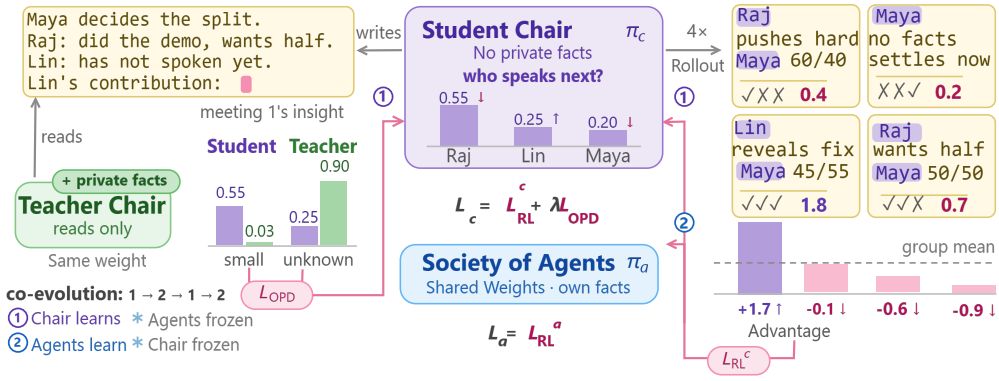



Figure 3: **CO-ADAPT co-evolves information routing and disclosure.** The chair policy $\pi_c$ learns whom to query via verifier-driven RL and privileged on-policy distillation (OPD), while the shared participant policy $\pi_a$ learns what private information to reveal via RL. Training alternates between chair and participant updates with the counterpart frozen, coupling routing and reasoning through verifiable content and provenance rewards.

To overcome the difficulty of recovering distributed information in Hidden Profile tasks (Stasser & Titus, 1985), we formulate the meeting as a sequential information-elicitation process with a chair that selects whom to query (§4.1) and introduce an asymmetric co-evolution procedure that learns routing and information disclosure from complementary signals (§4.4), as illustrated in Figure 3.

## 4 CO-ADAPT: CO-EVOLVING THE CHAIR AND THE PARTICIPANTS

We formulate the meeting as a sequential information-elicitation process in which a chair selects whom to query (§4.1), describe how the chair (§4.2) and the participants (§4.3) are trained from different signals, and introduce the alternating procedure that co-evolves the two (§4.4).

### 4.1 PROBLEM FORMULATION

We study the problem of training language-model agents to conduct meetings in which the information required for a correct decision is distributed across participants (Stasser & Titus, 1985). Each scenario contains a shared context, $N$ participants, and a set of decisive facts $z_i$ privately held by

participant $i$; one participant serves as the decision maker. We denote the complete set of decisive facts by $z_{1:N}$.

Following the Hidden Profile formulation of Li et al. (2025), we model the meeting as a sequential process, but make participant selection an explicit learned decision and evaluate the final settlement programmatically rather than with an LLM judge. At turn $t$, the chair observes a partial view $x_t$ containing the shared context, dialogue history, and its previous insight, and produces an insight $h_t$ and selects participant $i_t$; the selected participant then responds based on the same public view and its private facts. Formally, $(h_t, i_t) \sim \pi_c(\cdot \mid x_t)$, $y_t \sim \pi_a(\cdot \mid x_t, z_{i_t})$, and $x_{t+1} = \Phi(x_t, h_t, y_t)$, where $\pi_c$ is the chair policy, $\pi_a$ is a shared participant policy, and $\Phi$ updates the public state. The chair never observes decisive facts directly and can acquire them only when participants reveal them. The interaction terminates when the decision maker issues a settlement $\sigma$, yielding trajectory $\tau$.

We score each trajectory with a programmatic verifier. Let $V_\tau$ be the decisive facts revealed during $\tau$. The reward is $R(\tau) = \mathbf{1}[\text{settled}] \prod_{\phi \in C_{\text{con}}} \phi(\sigma) \prod_{\phi \in C_{\text{prov}}} \phi(\sigma, V_\tau)$, where $C_{\text{con}}$ contains content checks on the settlement and $C_{\text{prov}}$ contains provenance checks requiring the facts used by the settlement to have been explicitly revealed. Following rule-based verifiable rewards (Guo et al., 2025a), all checks are executed programmatically, so a correct settlement based on an unrevealed fact receives no credit. Our objective is to maximize $\mathbb{E}_\tau[R(\tau)]$. The two policies, however, face different information states and therefore different learning problems: the chair must infer whom to query without observing any decisive fact, while each participant already knows its own decisive facts and must learn what to reveal. We therefore train the chair and participants with complementary learning signals, described next.

### 4.2 Training the Chair

The chair faces an information-seeking problem: at each turn, it must infer what information is still missing from the public record and decide which participant is most likely to provide it. The participant-selection decision can therefore be learned directly from the verifier reward, whereas the chair's intermediate insight can benefit from an additional training-time signal. We use these two signals separately: verifier reward trains *where to look*, while privileged distillation helps the chair learn *what is missing* from its partial view.

**Routing from verifier reward.** For each scenario, we sample a group of $G$ trajectories while holding the participant policy fixed. Each trajectory receives a chair-specific reward that combines the verifier reward with routing credit for selecting participants who currently hold unrevealed decisive facts (Appendix C.2). We apply group-relative policy optimization (GRPO) (Shao et al., 2024), normalizing rewards within trajectories of the same scenario so that routing decisions are compared under matched task difficulty. The resulting advantage is applied only to the tokens corresponding to the chair's selection decisions. For negative-advantage trajectories, we further restrict the update to high-confidence erroneous tokens, following the confident-error formulation of (Wang et al., 2026a). The detailed advantage normalization and token-level objective are given in Appendix C.

**Privileged insight distillation.** Reward alone indicates whether a trajectory ultimately succeeds, but it does not directly reveal which information is missing from the chair's current view. This is especially important early in training, when the chair may not yet know how to infer where decisive information resides. We therefore introduce *privileged insight distillation*, inspired by on-policy distillation (Agarwal et al., 2024). During training, a teacher with access to all decisive facts provides a richer description of the current information state, while the student chair retains only the information available at inference time.

Concretely, the teacher is the same base chair model with its trainable adapter disabled, augmented with the complete set of decisive facts: $\pi_T(\cdot \mid x_t, z_{1:N})$. Teacher and student therefore share the same initialization and differ only in information access. Importantly, we distill the teacher's *insight*, rather than its routing decision. A privileged teacher can identify the participant holding a missing fact precisely because it already observes that fact; copying this decision would bypass the information- seeking problem faced by the student. Instead, the teacher summarizes the information missing from the chair's current view, allowing the student to learn to infer a similar information state from partial observations. The privileged signal is available only during training; at inference, the chair generates its insight from $x_t$ alone.

We apply OPD on the chair's own on-policy rollouts. For trajectories with positive routing advantage, the student's generated insight is used as the prefix on which the privileged teacher is queried, and the teacher's token distribution is distilled into the student:

$$\mathcal{L}_{\mathrm{OPD}} = \sum_{j:A_j^c>0} \sum_{t} \sum_{k=1}^{|h_t|} w_{t,k}\, D_{\mathrm{KL}}[\pi_T(\cdot \mid x_t, z_{1:N}, h_{t,<k}) \,\|\, \pi_c(\cdot \mid x_t, h_{t,<k})]\,, \tag{1}$$

where $h_{t,<k}$ denotes the student-generated prefix and $w_{t,k}$ is a token-level weight (Appendix C). Thus, OPD provides a richer training signal for learning to infer otherwise inaccessible information, while preserving the student's partial-information setting at inference time. The chair objective combines this privileged signal with the reward-based routing objective, $\mathcal{L}_c = \mathcal{L}_{\mathrm{RL}}^c + \lambda_{\mathrm{OPD}}\mathcal{L}_{\mathrm{OPD}}$, where $\lambda_{\mathrm{OPD}}$ controls the strength of distillation.

### 4.3 Training the Participants

All participants share a single policy $\pi_a$ and differ only in their decisive facts $z_i$, following parameter sharing in multi-agent reinforcement learning (Christianos et al., 2021). Their learning problem is complementary to the chair's: participants already possess the information relevant to the decision, but must learn when and how to disclose it so that it becomes useful to the society. We therefore train the shared participant policy using the verifier reward alone.

For each scenario, we sample $G$ trajectories with the chair policy held fixed and apply GRPO (Shao et al., 2024) to the participant responses and final settlement. Rewards are normalized within the scenario, and the resulting advantage is applied to the participant's generated response and settlement spans. Unlike the chair, participants receive no privileged signal because their private information is already directly available to them; the learning challenge is effective disclosure rather than inference of missing information. The complete participant objective and token-level details are provided in Appendix C.

### 4.4 Alternating Co-evolution

The two learning problems are coupled through their interaction: improving the chair changes which private information enters the public record, while improving participants changes the utility of subsequent queries. We therefore train them via alternating on-policy updates (Huang et al., 2026). In the chair phase, the participant policy is frozen, rollouts are collected, and the chair is updated using its reward-based routing objective and OPD. In the participant phase, the chair is frozen, new rollouts are collected, and the participant policy is updated using the verifier reward. Freezing the counterpart also keeps the within-scenario GRPO comparison well defined, since differences across the $G$ rollouts arise from the active policy alone. This alternating procedure enables co-adaptation: improved routing exposes more decisive information, while improved disclosure makes subsequent routing decisions more informative.

## 5 Experiments and Evaluation

### 5.1 Experimental Settings

**Baselines.** We compare Co-Adapt against three groups of baselines: large language models, existing multi-agent methods, and an untrained variant of our framework. For large language models, we evaluate GPT-5-Mini OpenAI (2025), Gemini-3.6-Flash Google DeepMind (2026), DeepSeek-V4.1-Flash DeepSeek-AI (2026), and Qwen3-235B-A22B-2507 Team (2025) without additional training. We additionally compare against PPDPP (Deng et al., 2024), Sotopia-RL (Yu et al., 2025), Sotopia-ToM (YS et al., 2026), SOTOPIA-$\Omega$ (Zhang et al., 2025b), and EPO (Liu et al., 2025a), using Qwen3-4B as the backbone. Finally, we compare Co-Adapt with a *Vanilla* variant that uses the same Qwen3-4B router and Qwen3-8B participant models but without co-evolutionary training. Thus, the comparison isolates the effect of training while keeping the underlying models and interaction environment fixed.

**Benchmarks.** We evaluate on ElicitBench, which contains 1,100 meeting scenarios spanning 11 domains. We use 720 scenarios from 9 domains for training and hold out 380 scenarios for

| Method | Seen (n = 180) | | | | | Unseen (n = 200) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Pass Rate ↑ | Solved ↑ | Content ↑ | Prov. ↑ | Turns | Pass Rate ↑ | Solved ↑ | Content ↑ | Prov. ↑ | Turns |
| *Large models* | | | | | | | | | | |
| GPT-5-Mini | 0.43 | 0.167 | 0.40 | 0.45 | 3.46 | 0.36 | 0.070 | 0.34 | 0.38 | 3.61 |
| Gemini-3.6-Flash | 0.49 | 0.189 | 0.44 | 0.54 | 3.42 | 0.52 | 0.185 | 0.49 | 0.58 | 3.51 |
| DeepSeek-V4.1-Flash | 0.50 | 0.196 | 0.45 | 0.50 | 3.44 | 0.52 | 0.189 | 0.50 | 0.53 | 3.51 |
| Qwen3-235B-A22B-2507 | 0.36 | 0.106 | 0.33 | 0.40 | 3.59 | 0.37 | 0.115 | 0.35 | 0.41 | 3.68 |
| *Baselines (Qwen3-8B)* | | | | | | | | | | |
| PPDPP (Deng et al., 2024) | 0.38 | 0.000 | 0.44 | 0.02 | 4.07 | 0.38 | 0.005 | 0.43 | 0.11 | 4.12 |
| Sotopia-RL (Yu et al., 2025) | 0.50 | 0.006 | 0.58 | 0.07 | 4.21 | 0.49 | 0.025 | 0.55 | 0.20 | 4.27 |
| Sotopia-ToM (YS et al., 2026) | 0.51 | 0.011 | 0.59 | 0.07 | 4.21 | 0.51 | 0.040 | 0.56 | 0.25 | 4.27 |
| SOTOPIA-Ω (Zhang et al., 2025b) | 0.49 | 0.000 | 0.57 | 0.04 | 4.21 | 0.48 | 0.010 | 0.55 | 0.20 | 4.27 |
| EPO (Liu et al., 2025a) | 0.44 | 0.017 | 0.46 | 0.31 | 4.11 | 0.43 | 0.025 | 0.43 | 0.38 | 4.16 |
| *Qwen3: 4B router, 8B agents* | | | | | | | | | | |
| Vanilla | 0.65 | 0.144 | 0.65 | 0.67 | 3.86 | 0.57 | 0.110 | 0.58 | 0.53 | 2.94 |
| **Co-Adapt** | **0.81** | **0.250** | **0.78** | **0.98** | 4.14 | **0.78** | **0.210** | **0.75** | **0.88** | 4.29 |
| *Gemma 3: 4B router, 12B agents* | | | | | | | | | | |
| Vanilla | 0.59 | 0.050 | 0.57 | 0.66 | 4.86 | 0.57 | 0.085 | 0.57 | 0.62 | 5.92 |
| **Co-Adapt** | **0.77** | **0.133** | **0.74** | **0.90** | 4.04 | **0.74** | **0.195** | **0.72** | **0.86** | 4.40 |
| *Phi: 4B router, 14B agents* | | | | | | | | | | |
| Vanilla | 0.72 | 0.161 | 0.69 | 0.89 | 6.24 | 0.74 | 0.135 | 0.71 | 0.85 | 6.16 |
| **Co-Adapt** | **0.78** | **0.172** | **0.75** | **0.96** | 4.02 | **0.78** | **0.150** | **0.75** | **0.93** | 4.25 |

Table 2: Results on the ELICITBENCH test set. Seen contains 180 held-out scenarios from the nine training domains; Unseen contains 200 scenarios from two domains never used in training. Vanilla uses the same Co-Adapt setup (router and agents) with untrained models. Additional backbones, per-domain results, and seed robustness are reported in Appendix D.

evaluation. The test set is divided into two groups: *Seen*, containing 180 held-out scenarios from the nine training domains, and *Unseen*, containing 200 scenarios from two domains that are never used during training. This split allows us to evaluate both performance on held-out scenarios from familiar domains and transfer to domains outside the training distribution.

**Training.** Our co-evolutionary framework uses a Qwen3-4B router and a Qwen3-8B participant model. The router and participant models are trained with LoRA adapters using bfloat16 precision. Additional details provided in Appendix C.

**Evaluation.** We evaluate each method on the Seen and Unseen splits using programmatic checks over the final settlement. We report five metrics: **Pass Rate**, **Solved**, **Content**, **Prov.**, and **Turns**. Pass Rate is the fraction of all content and provenance checks passed, while Solved is the fraction of scenarios in which every check passes and the episode settles. Content measures passed decision-field checks, averaging 6.92 per scenario in Seen and 6.90 in Unseen. Prov. measures passed provenance checks, averaging 1.34 and 1.44, respectively. Turns measures the number of turns until settlement, computed over settled episodes only. Since these programmatic metrics do not capture the quality of the interaction itself, we additionally conduct a human evaluation of routing quality, information disclosure, and decision quality on 100 held-out scenarios. The protocol and results are given in Appendix E.

| Method | Pass↑ | Solved↑ | Content↑ | Prov.↑ | Turns |
|---|---|---|---|---|---|
| *HiddenBench (Li et al., 2025)* | | | | | |
| Vanilla | 0.41 | 0.046 | 0.69 | 0.33 | 7.45 |
| **Co-Adapt** | **0.49** | **0.108** | **0.80** | **0.40** | **4.89** |

Table 3: Out-of-distribution results on HiddenBench ($n = 65$), with the Qwen3 4B router and 8B agents.

## 5.2 EXPERIMENTAL RESULTS AND ANALYSIS

**RQ1: Does CO-ADAPT improve multi-agent information elicitation and decision quality?** CO-ADAPT substantially improves both decision quality and information acquisition over the vanilla system. As shown in Table 2, on Qwen, Pass Rate increases from 0.65 to 0.81 on seen domains and from 0.57 to 0.78 on unseen domains, while Solved increases from 0.144 to 0.250 and from 0.110 to 0.210, respectively. The gains in both Content and Provenance further indicate that CO-

ADAPT does not merely produce better final decisions, but more effectively elicits and uses the private information required to support them.

**RQ2: Does CO-ADAPT generalize information-elicitation strategies to unseen domains?** The gains persist beyond the training distribution, suggesting that CO-ADAPT learns transferable information-elicitation strategies rather than domain-specific interaction patterns. Table 2 provides further evidence of this generalization, on unseen domains, Pass Rate improves from 0.57 to 0.78, with corresponding gains in Content (0.58 → 0.75) and Provenance (0.53 →0.88). This transfer is further supported by HiddenBench, an out-of-distribution evaluation (Table 3), where CO-ADAPT raises Pass Rate from 0.41 to 0.49 and Content from 0.69 to 0.80. These improvements in HiddenBench demonstrate that the learned elicitation behavior transfers effectively to a structurally different evaluation setting, improving partial verification even under out-of-distribution conditions.

**RQ3: Do the gains from CO-ADAPT reflect improved information acquisition and disclosure?** The improvement in Provenance provides direct evidence that the gains are tied to information acquisition rather than only final-response quality. Compared with Vanilla in Table 2, CO-ADAPT improves Provenance from 0.67 to 0.98 on seen domains and from 0.53 to 0.88 on unseen domains, while Content improves from 0.65 to 0.78 and from 0.58 to 0.75. Since provenance is satisfied only when the cited private facts were actually revealed during the interaction, these gains indicate that CO-ADAPT better coordinates participant selection and information disclosure.

| Variant | Pass Rate ↑ | Solved ↑ | Content ↑ | Prov. ↑ | Turns |
|---|---|---|---|---|---|
| *Training components* | | | | | |
| w/o Router RL | 0.60 | 0.111 | 0.60 | 0.61 | 3.78 |
| w/o OPD | 0.78 | 0.233 | 0.76 | 0.98 | 4.17 |
| w/o Agent RL | 0.72 | 0.133 | 0.72 | 0.78 | 4.04 |
| *Reward design* | | | | | |
| w/o routing credit | 0.79 | 0.244 | 0.77 | 0.95 | 4.13 |
| w/o penalties | 0.80 | 0.244 | 0.78 | 0.96 | 8.13 |
| w/o reveal progress | 0.79 | 0.244 | 0.76 | 0.94 | 4.30 |
| Outcome only (binary) | 0.75 | 0.200 | 0.73 | 0.88 | 4.10 |
| **Co-Adapt** | **0.81** | **0.250** | **0.78** | **0.98** | **4.14** |

Table 4: Ablations on the seen split of the 550-scenario setting (n = 90), with the Qwen3 4B router and 8B agents.

**RQ4: Does CO-ADAPT consistently improve performance across model backbones?** The improvement is consistent across both Qwen and Gemma, indicating that it is not specific to a single backbone. Table 2 shows, Pass Rate increases from 0.65 to 0.81 for Qwen and from 0.59 to 0.77 for Gemma3 on seen domains, with similar gains on unseen domains (0.57→0.78 and 0.57→0.74). The consistent improvements across Solved, Content, and Provenance further suggest that CO-ADAPT improves the underlying elicitation process rather than a backbone-specific output behavior.

### 5.3 ABLATION STUDIES

**Impact of training components.** Table 4 shows that removing Router RL reduces Pass Rate from 0.81 to 0.60 and Provenance from 0.98 to 0.61, while removing Agent RL lowers Pass Rate to 0.72, Solved to 0.133, and Provenance to 0.78. Thus, both routing and disclosure are necessary for effective information elicitation. **Impact of reward design.** As shown in Table 4, using only a binary outcome reward lowers Pass Rate from 0.81 to 0.75 and Provenance from 0.98 to 0.88. Routing credit and reveal progress further improve elicitation, while penalties primarily improve efficiency: removing them increases average turns from 4.14 to 8.13 with little change in final performance. Overall, the ablations support the complementary roles of co-learning routing and disclosure and of structured reward shaping.

## 6 CONCLUSION

We presented ELICITBENCH, a verifiable benchmark for evaluating information elicitation and decision making under distributed private information, and CO-ADAPT, a co-evolutionary framework that jointly optimizes participant reasoning and information-aware routing. Across held-out domains, CO-ADAPT consistently improves both decision quality and provenance over vanilla multi-agent interaction, with further gains on out-of-distribution HiddenBench scenarios. Our analysis shows that these improvements are accompanied by more precise routing and more effective information disclosure, while remaining consistent across model backbones. Together, these results highlight the importance of jointly learning whom to query and what information to reveal for reliable multi-agent decision making under hidden information.

## REFERENCES


Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, 2024.

Salaheddin Alzubi, Noah Provenzano, Jaydon Bingham, Weiyuan Chen, and Tu Vu. Evoskill: Automated skill discovery for multi-agent systems. *arXiv preprint arXiv:2603.02766*, 2026.

Weize Chen, Jiarui Yuan, Chen Qian, Cheng Yang, Zhiyuan Liu, and Maosong Sun. Optima: Optimizing effectiveness and efficiency for llm-based multi-agent system. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11534–11557, 2025a.

Yixing Chen, Yiding Wang, Siqi Zhu, Haofei Yu, Tao Feng, Muhan Zhang, Mostofa Patwary, and Jiaxuan You. Multi-agent evolve: Llm self-improve through co-evolution. *arXiv preprint arXiv:2510.23595*, 2025b.

Filippos Christianos, Georgios Papoudakis, Muhammad A Rahman, and Stefano V Albrecht. Scaling multi-agent reinforcement learning with selective parameter sharing. In *International conference on machine learning*, pp. 1989–1998. PMLR, 2021.

Herbert H Clark and Susan E Brennan. Grounding in communication. 1991.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

Yufan Dang, Chen Qian, Xueheng Luo, Jingru Fan, Zihao Xie, Ruijie Shi, Weize Chen, Cheng Yang, Xiaoyin Che, Ye Tian, et al. Multi-agent collaboration via evolving orchestration. *Advances in neural information processing systems*, 38:165025–165059, 2026.

DeepSeek-AI. Deepseek-v4.1-flash: Pushing the limits of kv cache compression, 2026.

Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. Plug-and-play policy planner for large language model powered dialogue agents. In *International Conference on Learning Representations*, volume 2024, pp. 10058–10090, 2024.

GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang,

Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Google DeepMind. Gemini 3.6 Flash model card. https://deepmind.google/models/model-cards/gemini-3-6-flash/, July 2026. Accessed: 2026-09-26.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025a.

Jiacheng Guo, Ling Yang, Peter Chen, Qixin Xiao, Yinjie Wang, Xinzhe Juan, Jiahao Qiu, Ke Shen, and Mengdi Wang. Genenv: Difficulty-aligned co-evolution between llm agents and environment simulators. *arXiv preprint arXiv:2512.19682*, 2025b.

Yihao Hu, Zhihao Wen, Xiujin Liu, Pan Wang, Xin Zhang, and Wei Wu. Seal: Synergistic co-evolution of agents and learning environments. *arXiv preprint arXiv:2605.24426*, 2026.

Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-zero: Self-evolving reasoning llm from zero data. In *International Conference on Learning Representations*, volume 2026, pp. 130770–130790, 2026.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Eric Hanchen Jiang, Levina Li, Rui Sun, Xiao Liang, Yubei Li, Yuchen Wu, Haozheng Luo, Hengli Li, Zhi Zhang, Zhaolu Kang, et al. Agent q-mix: Selecting the right action for llm multi-agent systems through reinforcement learning. *arXiv preprint arXiv:2604.00344*, 2026.

Zixuan Ke, Yifei Ming, Austin Xu, Ryan Chin, Xuan-Phi Nguyen, Prathyusha Jwalapuram, Jiayu Wang, Semih Yavuz, Caiming Xiong, and Shafiq Joty. Mas-orchestra: Understanding and improving multi-agent reasoning through holistic orchestration and controlled benchmarks. *arXiv preprint arXiv:2601.14652*, 2026.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. Fantom: A benchmark for stress-testing machine theory of mind in interactions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14397–14413, 2023.

James R Larson, Pennie G Foster-Fishman, and Christopher B Keys. Discussion of shared and unshared information in decision-making groups. *Journal of personality and social psychology*, 67(3):446, 1994.

James R Larson Jr, Pennie G Foster-Fishman, and Timothy M Franz. Leadership style and the discussion of shared and unshared information in decision-making groups. *Personality and Social Psychology Bulletin*, 24(5):482–495, 1998.

Yuxuan Li, Aoi Naito, and Hirokazu Shirado. Systematic failures in collective reasoning under distributed information in multi-agent llms. *arXiv preprint arXiv:2505.11556*, 2025.

Wei Liu, Chenxi Wang, Yifei Wang, Zihao Xie, Rennai Qiu, Yufan Dang, Zhuoyun Du, Weize Chen, Cheng Yang, and Chen Qian. Autonomous agents for collaborative task under information asymmetry. *Advances in Neural Information Processing Systems*, 37:2734–2765, 2024.

Xiaoqian Liu, Ke Wang, Yongbin Li, Yuchuan Wu, Wentao Ma, Aobo Kong, Fei Huang, Jianbin Jiao, and Junge Zhang. Epo: Explicit policy optimization for strategic reasoning in llms via reinforcement learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15371–15396, 2025a.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025b.

Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. A dynamic llm-powered agent network for task-oriented agent collaboration. *arXiv preprint arXiv:2310.02170*, 2023.

Li Lu, Y Connie Yuan, and Poppy Lauretta McLeod. Twenty-five years of hidden profiles in group decision making: A meta-analysis. *Personality and Social Psychology Review*, 16(1):54–75, 2012.

Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, et al. Self-distilled agentic reinforcement learning. *arXiv preprint arXiv:2605.15155*, 2026.

Hao Ma, Tianyi Hu, Zhiqiang Pu, Boyin Liu, Xiaolin Ai, Yanyan Liang, and Min Chen. Coevolving with the other you: Fine-tuning llm with sequential cooperative multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 37:15497–15525, 2024.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can llms keep a secret? testing privacy implications of language models via contextual integrity theory. In *International Conference on Learning Representations*, volume 2024, pp. 1892–1915, 2024.

Sumeet Ramesh Motwani, Chandler Smith, Rocktim Jyoti Das, Rafael Rafailov, Ivan Laptev, Philip HS Torr, Fabio Pizzati, Ronald Clark, and Christian Schroeder de Witt. Malt: Improving reasoning with multi-agent llm training. *arXiv preprint arXiv:2412.01928*, 2024.

OpenAI. GPT-5 system card. `https://cdn.openai.com/gpt-5-system-card.pdf`, August 2025. Accessed: 2026-09-26.

Chanwoo Park, Seungju Han, Xingzhi Guo, Asuman E Ozdaglar, Kaiqing Zhang, and Joo-Kyung Kim. Maporl: Multi-agent post-co-training for collaborative large language models with reinforcement learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30215–30248, 2025.

Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models. *arXiv preprint arXiv:2602.04942*, 2026.

Run Peng, Ziqiao Ma, Amy Pang, Sikai Li, Zhang Xi-Jia, Yingzhuo Yu, Cristian-Paul Bara, and Joyce Chai. Communication and verification in llm agents towards collaboration under information asymmetry. *arXiv preprint arXiv:2510.25595*, 2025.

Giorgio Piatti, Zhijing Jin, Max Kleiman-Weiner, Bernhard Schölkopf, Mrinmaya Sachan, and Rada Mihalcea. Cooperate or collapse: Emergence of sustainable cooperation in a society of llm agents. *Advances in Neural Information Processing Systems*, 37:111715–111759, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Garold Stasser and William Titus. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of personality and social psychology*, 48 (6):1467, 1985.

Garold Stasser and William Titus. Effects of information load and percentage of shared information on the dissemination of unshared information during group discussion. *Journal of personality and social psychology*, 53(1):81, 1987.

Garold Stasser and William Titus. Hidden profiles: A brief history. *Psychological inquiry*, 14(3-4): 304–313, 2003.

Garold Stasser, Dennis D Stewart, and Gwen M Wittenbaum. Expert roles and information exchange during discussion: The importance of knowing who knows what. *Journal of experimental social psychology*, 31(3):244–265, 1995.

Garold Stasser, Sandra I Vaughan, and Dennis D Stewart. Pooling unshared information: The benefits of knowing how access to information is distributed among group members. *Organizational behavior and human decision processes*, 82(1):102–116, 2000.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Aozhe Wang, Zhengxi Lu, Jianze Wang, Shangke Lv, Ying Liu, Weiming Lu, Jun Xiao, Yueting Zhuang, Hua Yang, Qianglong Chen, et al. Ttpo: Test-time policy optimization. *arXiv preprint arXiv:2608.27448*, 2026a.

Zi-Han Wang, Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Jie Wu, Zhengzhou Cai, Yueqing Sun, Ziang Ye, Linji Hao, Qi Gu, et al. Agentopsd: Recursive self-distillation for agentic reinforcement learning. *arXiv preprint arXiv:2608.05987*, 2026b.

Daniel M Wegner. Transactive memory: A contemporary analysis of the group mind. In *Theories of group behavior*, pp. 185–208. Springer, 1987.

Jinyang Wu, Shuo Yang, Zhengxi Lu, Fan Zhang, Yuhao Shen, Lang Feng, Haoran Luo, Zheng Lian, Shuai Zhang, Zhengqi Wen, et al. Seed: Self-evolving on-policy distillation for agentic reinforcement learning. *arXiv preprint arXiv:2607.14777*, 2026a.

Xiyang Wu, Zongxia Li, Guangyao Shi, Alexander Duffy, Tyler Marques, Matthew Lyle Olson, Tianyi Zhou, and Dinesh Manocha. Co-evolving llm decision and skill bank agents for long-horizon tasks. *arXiv preprint arXiv:2604.20987*, 2026b.

Xiangyuan Xue, Yifan Zhou, Guibin Zhang, Zaibin Zhang, Yijiang Li, Chen Zhang, Zhenfei Yin, Philip Torr, Wanli Ouyang, and Lei Bai. Comas: Co-evolving multi-agent systems via interaction rewards. In *International Conference on Learning Representations*, volume 2026, pp. 32373–32394, 2026.

Shidong Yang, Ziyu Ma, Tongwen Huang, Yiming Hu, Yong Wang, and Xiangxiang Chu. Coevolve: Training llm agents via agent-data mutual evolution. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23015–23036, 2026.

Yashwanth YS, Ruichen Wang, Shihua Zeng, Xuhui Zhou, Koichi Onoue, Vasudha Varadarajan, and Maarten Sap. Sotopia-tom: Evaluating information management in multi-agent interaction with theory of mind. *arXiv preprint arXiv:2605.02307*, 2026.

Haofei Yu, Zhengyang Qi, Yining Zhao, Kolby Nottingham, Keyang Xuan, Bodhisattwa Prasad Majumder, Hao Zhu, Paul Pu Liang, and Jiaxuan You. Sotopia-rl: Reward design for social intelligence. *arXiv preprint arXiv:2508.03905*, 2025.

Yanwei Yue, Guibin Zhang, Boyang Liu, Guancheng Wan, Kun Wang, Dawei Cheng, and Yiyan Qi. Masrouter: Learning to route llms for multi-agent systems. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15549–15572, 2025.

Hanrong Zhang, Shicheng Fan, Henry Peng Zou, Yankai Chen, Zhenting Wang, Jiayu Zhou, Chengze Li, Wei-Chieh Huang, Yifei Yao, Kening Zheng, et al. Coevoskills: Self-evolving agent skills via co-evolutionary verification. *arXiv preprint arXiv:2604.01687*, 2026a.

Haozhen Zhang, Tao Feng, and Jiaxuan You. Router-r1: Teaching llms multi-round routing and aggregation via reinforcement learning. *Advances in Neural Information Processing Systems*, 38: 141233–141265, 2026b.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, et al. Aflow: Automating agentic workflow generation. In *International Conference on Learning Representations*, volume 2025, pp. 34040–34077, 2025a.

Wenyuan Zhang, Tianyun Liu, Mengxiao Song, Xiaodong Li, and Tingwen Liu. Sotopia-■: Dynamic strategy injection learning and social instruction following evaluation for social agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24669–24697, 2025b.

Andrew Zhao, Yiran Wu, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *Advances in Neural Information Processing Systems*, 38:105816–105879, 2026a.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026b.

Yujie Zhao, Lanxiang Hu, Yang Wang, Minmin Hou, Hao Zhang, Ke Ding, and Jishen Zhao. Stronger-mas: Multi-agent reinforcement learning for collaborative llms. In *International Conference on Learning Representations*, volume 2026, pp. 150619–150651, 2026c.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.

Xuhui Zhou, Zhe Su, Tiwalayo Eisape, Hyunwoo Kim, and Maarten Sap. Is this the real life? is this just fantasy? the misleading success of simulating social interactions with llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21692–21714, 2024a.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. Sotopia: Interactive evaluation for social intelligence in language agents. In *International Conference on Learning Representations*, volume 2024, pp. 40975–41019, 2024b.

Mingchen Zhuge, Wenyi Wang, Louis Kirsch, Francesco Faccio, Dmitrii Khizbullin, and Jürgen Schmidhuber. Language agents as optimizable graphs. *arXiv preprint arXiv:2402.16823*, 2024.

# A Appendix

The Appendix provides supplementary material, including the AI use statement A.1, ethics statement A.2, reproducibility statement A.3, dataset construction B, training and reward details C, additional experimental results D, human evaluation E, and prompts used F.

## A.1 AI use statement

We used large language models (LLMs) to assist with manuscript writing and language polishing, and for literature retrieval and discovery of related work. LLMs served only as supporting tools; all research decisions, methodology, experiments, analyses, and conclusions were conducted and verified by the authors. Further details are provided in the relevant sections of the paper.

## A.2 Ethics Statement

Our scenario seeds are derived from accounts provided by 53 practitioners and members of the public across professional and everyday decision-making domains. Respondents described past decision failures or near-failures from memory, with names, places, and figures altered; no identifying information was collected. We retained only accounts satisfying the study's information-distribution criteria and used these accounts as scenario seeds for controlled model experiments. The resulting system is evaluated as a research framework for information elicitation and decision-making rather than as an autonomous decision-maker in real-world high-stakes settings.

## A.3 Reproducibility Statement

We have taken careful steps to ensure the reproducibility of our work. At this stage, we have created an anonymous repository (`https://anonymous.4open.science/r/ICLR777/README.md`) containing sample benchmark data and the code for our main method. Upon acceptance, we plan to release the complete implementation, benchmark details, and a dedicated project page to facilitate full reproducibility and further research.

# B Dataset Construction

## B.1 Scenario Specification

Each scenario in ElicitBench describes a group of three to five agents who must jointly reach a single decision. Exactly one agent is designated the *decision-maker* and holds final authority. Every other agent holds one or more *private facts* that are unknown to the decision-maker: an unrecorded clinical finding, a deadline known only to a clerk, a family circumstance disclosed at intake. Each private fact is *decisive*, in the sense that the correct decision changes once it is known. If the private facts remain unspoken, the decision-maker chooses incorrectly; if they are surfaced in conversation, the group can choose correctly.

A scenario comprises the following components.

- **Shared context.** A short list of facts visible to every agent, written as a participant would actually observe them and containing no indication that information is missing. One shared fact always supplies an ordinary reason for every agent to speak before the decision is final, such as a routine check-in or a standing policy.
- **Private facts.** Each is owned by exactly one non-decision-maker and never appears in any other agent's view. Every non-decision-maker owns at least one.
- **Views.** The information available to each agent: the shared context together with that agent's own private facts and nothing else.
- **Settlement schema.** The structured output the decision-maker must produce: the decisions taken, the facts credited, any commitments made, the facts cited as justification, and the facts revealed during the interaction.

- **Checks.** *Content checks* test whether the decision is correct, with thresholds taken verbatim from the private facts. *Provenance checks* test whether the relevant private facts were actually revealed, so that a group cannot pass by guessing. A settlement passes only if every check is satisfied.

### B.2 SCENARIO STRUCTURE

Figure 1 illustrates the structure of an ELICITBENCH scenario. Consider a team that must divide a $30,000 bonus among Maya, Raj, and Lin. Maya is the team lead and has the authority to set the split. Raj is a senior engineer who led the visible product demo and privately believes he deserves at least half of the bonus. Lin is a junior engineer who privately knows that she fixed a database crash that could have sunk the launch and that she has a competing job offer; unlike Raj, she primarily wants her contribution to be recognized. No participant observes all of these facts: Maya does not know about Lin's hidden contribution and exit risk and Raj does not know about Lin's contribution. The final agreement is checked against concrete requirements: it must acknowledge Lin's database fix, give Lin a fair share, and credit Raj's visible work. These requirements cannot be satisfied from Maya's initial view alone because the information needed to determine the appropriate agreement is distributed across the participants. The scenario can succeed only if the relevant private facts are surfaced during interaction and incorporated into the final agreement.

**Participants.** Each participant has a name and a role. The role provides a cue about who is likely to know which information. In human groups, members direct questions to the person whose role suggests they hold the relevant information (Wegner, 1987). The decision maker in ELICITBENCH must similarly use the participants' roles to determine whom to engage.

**Shared context.** The shared context describes the situation, the decision to be made, who has the authority to make it, and the process the group normally follows. Each scenario has between 4 and 12 shared facts (mean 5.0). It describes the world rather than the information-acquisition task: it does not say or imply that some information is missing. The shared context also includes one ordinary process step, such as a weekly review meeting, that gives every participant a natural reason to speak before the decision is made.

**Decision and verification.** The settlement is evaluated entirely programmatically. *Content checks* evaluate the settlement itself, while *provenance checks* additionally verify that required private facts were actually revealed and cited. Each check is a boolean expression over the settlement and revealed facts, with no LLM-based judge involved in scoring.

**Interaction.** An episode is one run of a scenario in which participants take turns writing to a shared public record. The decision maker can either `say` or `settle`, while other participants can `say` or `reveal` their private fact. A fact is considered elicited only when its owner uses `reveal`; mentioning it in `say` does not count (Clark & Brennan, 1991). The decision maker can settle only after every other participant has had at least one turn.

**Domains.** We selected eleven domains to cover a broad range of settings in which agents may need to exchange private information to reach a decision. Each domain is specified by a setting, typical decision-maker roles, scenario types, and decision types.

The released dataset contains 1,100 scenarios across 11 domains (100 per domain): healthcare, software and technology, defense, finance, education, entertainment, legal, manufacturing, workplace interpersonal, friends and family, and informal commerce and bargaining. Agent counts are balanced across three, four and five agents (360, 369 and 371 scenarios respectively).

### B.3 AUTOMATED GENERATION AND CERTIFICATION

Scenarios are produced by a loop of language models, each with a fixed role. A scenario is accepted only after it has passed every gate described below.

**Challenger.** A GPT-class model writes a candidate scenario. Its input consists of the rules a scenario must obey, a short description of the domain, the required number of agents, and the descriptions of the twelve most recent scenarios in that domain with an instruction not to repeat them. The rules require exactly one decision-maker, at least one decisive private fact for every other agent, no

private fact appearing in another agent's view, no sentence in the shared context that hints at hidden information, and one ordinary reason for every agent to speak. Checks may refer only to the fields of the settlement schema and must use exact values from the private facts; softened thresholds are prohibited.

**Verifier.** The same model examines the candidate statically and answers three questions. *Leakage*: could the decision-maker satisfy every check using only the shared context and ordinary inference? *Satisfiability*: does at least one settlement satisfy every check? *Falsifiability*: does at least one plausible settlement fail a check? A scenario failing any of these is rejected as *malformed*.

**Lone Agent Arm.** A stronger model GLM 5.2 GLM-5-Team et al. (2026) plays the decision-maker in isolation. It receives only the decision-maker's view and produces a settlement immediately, without conversation. Four independent attempts are made at sampling temperatures drawn from the interval $[0.2, 0.6]$. The scenario passes this gate only if at most one of the four attempts satisfies the checks; otherwise it is solvable without the group and is rejected as *leaked*.

**Society of Agents Arm.** A stronger model GLM 5.2 plays every agent in a full multi-turn conversation. Each agent has access only to its own view, agents speak in a fixed order up to a turn cap, and the decision-maker settles at the end. Four independent conversations are run under the same temperature range. The scenario passes only if at least three of the four satisfy the checks; otherwise the group cannot solve it either, and the scenario is rejected as *uncoordinated*.

**Revision loop.** A rejected scenario is not discarded. A language model reads the failure and writes a brief diagnosis together with fix instructions; the Challenger rewrites the scenario accordingly, and the revised scenario re-enters the pipeline at the Verifier. A scenario may be rejected up to twenty times at any single gate before it is abandoned. Over the 1,100 accepted scenarios, acceptance required 5.5 rounds on average (median 4, maximum 33); 185 scenarios required more than eight rounds.

Taken together, the Verifier and the two arms confer a certificate on each accepted scenario: it is well-formed, it cannot be solved by the decision-maker alone, and it can be solved by the coordinating group.

### B.4 Human-in-the-Loop Checkpoints

A pipeline in which models judge models has an evident weakness: the hidden facts are whatever the Challenger imagines people fail to say, and no person has confirmed that the scenarios are realistic, that the checks are faithful, or that the model judges are accurate. We therefore introduce four human checkpoints. Human annotation and auditing at the validation checkpoints were conducted by postgraduate students, who were compensated according to the standard payment norms of our university.

**H1: Practitioner-sourced seeds.** We asked practitioners with at least two years of experience in each domain (nurses, case managers, engineers, paralegals, shift supervisors, and members of the public for the everyday domains) to describe a specific occasion on which a decision at work went wrong, or nearly did, because someone present knew something relevant and did not say it in time. Each respondent answered six short prompts: their role and experience, the setting and who held final authority, what the decision-maker could see, what each other person knew and why it mattered, why it was not said, and what the correct decision turned out to be. Respondents wrote from memory with names, places and figures altered, so no identifying information was collected, and consented to the release of their account. An account was retained only if it featured one decision-maker, at least two other individuals holding distinct facts, each fact altering the outcome in a concrete way. Approximately half of the accounts met this standard. In total, 53 practitioners contributed 45 usable seeds.

Each seed was supplied to the Challenger as the situation to be compiled, with instructions to preserve the decision, the distribution of information and the reasons for silence, to alter identifying details, and not to introduce hidden facts the practitioner had not reported. Each seed yielded two or three structurally distinct scenarios at three, four and five agents, obtained by removing or splitting

a fact. These scenarios passed through the same gates as all others, with one additional Verifier question: does every private fact correspond to something the practitioner reported?

**H2: Human editor for stalled revisions.** When a scenario reaches its ninth revision round, or would otherwise be abandoned, the model diagnostic is disabled and a human editor assumes the role. The editor reviews the scenario, the eight transcripts and the preceding diagnoses, identifies the underlying cause (in one instance a turn cap so low that the final speaker never received a second turn), and either writes the fix instructions, edits the scenario directly, or discards it. The editor's instructions enter the Challenger precisely as the model's would, and the round is recorded as human-authored. A total of 185 scenarios (17%) received human revision.

**H3: Calibration of the model judge.** To estimate the reliability of the Verifier, we sampled 200 of its verdicts, stratified as 100 passes and 100 rejections and distributed across the domains with at most one verdict per scenario. Two annotators answered the Verifier's three questions for each scenario, blind to the Verifier's verdict, and assigned their own pass or reject label (Cohen's $\kappa = 0.75$; Cohen, 1960); disagreements were resolved by discussion. The Verifier agreed with the human labels on 83% of scenarios; it passed 7% of scenarios the annotators judged to leak and rejected 23% that they judged sound. No scenarios were altered during the procedure.

**H4: Human audit gate.** No scenario enters the dataset on model approval alone. Three raters scored each audited scenario on five criteria: whether the situation is realistic; whether the shared context reads naturally with no cue that information is hidden; whether every check faithfully encodes its private fact without a softened threshold; whether the private fact is one a person would plausibly not volunteer; and whether the decision is well-posed. Every seeded scenario was audited, together with twenty scenarios per domain from the remainder. Inter-rater reliability, measured by Krippendorff's $\alpha$, was 0.81. A scenario accepted by a majority of raters is marked *human-verified*. A scenario rejected by a majority is returned to the revision loop with the raters' comments as fix instructions and is audited again following re-acceptance.

### B.5 Certification Criteria

A scenario is released only after it has acquired three stamps in sequence:

1. **V (well-formed)**, conferred by the Verifier;
2. **G (needs the group)**, conferred jointly by the two arms: the scenario is solved in at most one of four attempts by the lone decision-maker and in at least three of four conversations by the group;
3. **H (human-verified)**, conferred by majority agreement of three raters.

Figure 2 depicts the pipeline with the human checkpoints indicated.

### B.6 Dataset Statistics

Table 5 summarizes the final dataset by domain. "Types" denotes the number of unique scenario types after merging near-synonymous labels. "3/4/5" gives the number of three-, four-, and five-agent scenarios. "Shared", "Content", and "Prov." are the average numbers of shared facts, content checks, and provenance checks per scenario. "Avg. Rounds" is the rounds-to-convergence averaged within each unique scenario type before averaging across the domain, and "Mean Cap" is the average interaction turn cap.

## C Training and Reward Details

This section provides implementation details for Co-Adapt that are omitted from the main paper for brevity. We describe the reward construction and alternating optimization procedure.

| Domain | $n$ | Types | 3/4/5 | Shared | Content | Prov. | Rounds | Cap |
|---|---|---|---|---|---|---|---|---|
| Healthcare | 100 | 34 | 33/34/33 | 4.95 | 5.52 | 1.24 | 3.95 | 15.09 |
| Software & Technology | 100 | 31 | 34/33/33 | 4.82 | 7.54 | 1.24 | 4.66 | 15.29 |
| Defense | 100 | 35 | 33/33/34 | 5.13 | 6.68 | 1.19 | 5.80 | 14.44 |
| Finance | 100 | 49 | 32/34/34 | 4.83 | 7.07 | 1.41 | 4.38 | 14.80 |
| Education | 100 | 39 | 33/33/34 | 4.87 | 7.28 | 1.23 | 5.39 | 15.09 |
| Entertainment | 100 | 56 | 32/34/34 | 4.64 | 7.64 | 1.24 | 5.67 | 15.58 |
| Legal | 100 | 30 | 33/33/34 | 4.54 | 6.82 | 1.18 | 5.81 | 15.49 |
| Manufacturing | 100 | 49 | 33/34/33 | 4.88 | 7.10 | 1.18 | 4.75 | 15.45 |
| Workplace Interpersonal | 100 | 38 | 32/34/34 | 5.27 | 6.59 | 1.90 | 6.30 | 15.86 |
| Friends, Family & Informal | 100 | 15 | 33/33/34 | 5.37 | 6.63 | 1.30 | 7.54 | 15.42 |
| Commerce / Bargaining | 100 | 33 | 32/34/34 | 5.55 | 7.21 | 1.57 | 8.02 | 15.78 |
| All | 1,100 | 409 | 360/369/371 | 4.99 | 6.92 | 1.33 | 5.50 | 15.30 |

Table 5: Per-domain statistics of ELICITBENCH. Columns: number of scenarios ($n$), distinct scenario types, split across 3-, 4- and 5-agent settings, mean shared / content / provenance items, mean dialogue rounds, and mean capacity.

### C.1 REWARD CONSTRUCTION

We distinguish between the *official* benchmark reward and the shaped reward used during training. The official reward is binary and is used for evaluation. Let $V_\tau$ denote the set of private facts formally revealed during trajectory $\tau$, i.e., facts revealed through a valid `reveal` action by their owner. Importantly, $V_\tau$ may contain both decisive and non-decisive private facts.

The official reward is

$$R_{\text{off}}(\tau) = \mathbb{I}[\text{settled}]\mathbb{I}[|\mathcal{C}_{\text{con}}| + |\mathcal{C}_{\text{prov}}| > 0] \prod_{\phi \in \mathcal{C}_{\text{con}}} \phi(\sigma) \prod_{\phi \in \mathcal{C}_{\text{prov}}} \phi(\sigma, V_\tau), \tag{2}$$

where $\sigma$ is the final settlement. The content checks $\mathcal{C}_{\text{con}}$ evaluate the settlement itself, while provenance checks $\mathcal{C}_{\text{prov}}$ verify that the private facts used by the settlement were formally revealed during the interaction. All checks are executed programmatically.

For policy optimization, we use a denser shaped reward. Let $c(\tau)$ denote the fraction of content checks passed, $p(\tau)$ the fraction of provenance checks passed, and $d(\tau)$ the fraction of decisive private facts that have been revealed. The training reward is

$$\begin{aligned} R_{\text{train}}(\tau) = {} & w_S R_{\text{off}}(\tau) + w_C c(\tau) + w_P p(\tau) + w_D d(\tau) \\ & - \min(\lambda_{\text{inv}} N_{\text{inv}}(\tau), \lambda_{\text{inv}}^{\max}) - \lambda_{\text{free}} N_{\text{free}}(\tau) \\ & - \lambda_{\text{turn}} \max(0, T(\tau) - T_{\min}(\tau)) - \lambda_{\text{timeout}} \mathbb{I}[\neg\text{settled}], \end{aligned} \tag{3}$$

where $N_{\text{inv}}$ counts invalid participant actions and $N_{\text{free}}$ counts unnecessary `free` actions.

Specifically, $N_{\text{inv}}$ includes participant actions that are unparseable, illegal in the current state, empty settlements, or attempts to reveal an unknown, foreign, or already-revealed fact. Invalid router selections are logged separately and are not included in $N_{\text{inv}}$.

The quantity $N_{\text{free}}$ counts `free` actions taken by a participant who still holds at least one unrevealed private fact of their own. This condition is not restricted to decisive facts: any unrevealed private fact makes the `free` action unnecessary for purposes of this penalty.

The turn threshold is

$$T_{\min}(\tau) = N_{\text{decisive}} + 1, \tag{4}$$

which is the minimum number of turns needed to reveal each decisive fact once and then settle; only turns beyond this threshold are penalized.

The coefficients used in our experiments are

$$\begin{aligned} w_S &= 1.00, & w_C &= 0.50, & w_P &= 0.20, & w_D &= 0.15, \\ \lambda_{\text{inv}} &= 0.10, & \lambda_{\text{inv}}^{\max} &= 0.30, & \lambda_{\text{free}} &= 0.02, \\ \lambda_{\text{turn}} &= 0.01, & \lambda_{\text{timeout}} &= 0.10. \end{aligned} \tag{5}$$

The content and provenance terms are zero for an unsettled episode, even if the episode has accumulated partial information. The reveal-progress term $d(\tau)$ is still credited based on the decisive facts revealed during the episode. Thus, an episode that fails to settle can nevertheless receive a nonzero reveal-progress reward.

### C.2 Routing Credit

In addition to the verifier-based reward, the chair receives a routing credit for selecting a participant that currently holds an unrevealed decisive fact. Let $H_t$ denote the set of participants that hold at least one decisive fact that has not yet been revealed before the participant acts at turn $t$. If the chair selects participant $i_t$, the routing hit is

$$h_t = \mathbb{I}[i_t \in H_t]. \tag{6}$$

The resulting chair reward is

$$R_c(\tau) = R_{\text{train}}(\tau) + \lambda_{\text{route}} \sum_{t=1}^{T(\tau)} h_t, \tag{7}$$

where $\lambda_{\text{route}} = 0.05$.

The routing credit is based on whether the selected participant currently holds an unrevealed decisive fact; it does not require that the fact be revealed on that turn. Consequently, selecting the same participant on multiple turns can produce multiple routing hits if that participant continues to hold an unrevealed decisive fact. The resulting credit is therefore bounded by the interaction constraints, including the per-participant speaking budget and the overall episode turn limit, rather than by the number of decisive facts.

The routing credit is included in the scalar chair reward on every turn. Separately, forced hand-offs, fallback selections following invalid router outputs, and turns with only one legal participant are excluded from the routing-policy gradient. These exclusions affect the policy-gradient mask rather than the computation of $R_c(\tau)$.

Participants do not receive routing credit and instead use

$$R_a(\tau) = R_{\text{train}}(\tau). \tag{8}$$

### C.3 Group-Relative Policy Optimization

For each scenario, we sample $G = 4$ independent trajectories using the same counterpart policy. During a router update, the participant policy is frozen; during a participant update, the router is frozen. Let $r_j$ denote the relevant reward for trajectory $j$ in a group. We compute the group-relative advantage using the group mean and sample standard deviation:

$$A_j = \frac{r_j - \mu_G}{s_G + \epsilon}, \qquad \mu_G = \frac{1}{G} \sum_{k=1}^{G} r_k, \tag{9}$$

where

$$s_G = \sqrt{\frac{1}{G-1} \sum_{k=1}^{G} (r_k - \mu_G)^2} \tag{10}$$

and $\epsilon = 10^{-4}$. Advantages are clipped to the interval $[-5, 5]$. If all trajectories in a group receive the same reward, the group is treated as degenerate and all advantages are set to zero.

The use of within-scenario groups ensures that the policy is compared against alternative rollouts of the same underlying scenario rather than against trajectories from scenarios with different intrinsic difficulty. The main paper uses this group-relative estimator for both the chair and participant policies.

For a scored generation span $y = (y_1, \ldots, y_L)$, the policy-gradient loss for one rollout is

$$\mathcal{L}_{\mathrm{RL}}(y) = -A \frac{1}{L} \sum_{k=1}^{L} \log \pi_\theta(y_k \mid y_{<k}, x). \tag{11}$$

The token average makes each decision a single learning unit rather than allowing longer generations, such as settlements, to dominate the gradient solely through their length.

For the chair, the scored span is only the generated `next_agent` value, rather than the accompanying textual insight. The participant objective scores the generated textual response, including the settlement text when the decision maker settles. Thus, the routing and participant policies receive credit on the parts of their outputs that directly determine the corresponding decision.

### C.4 Token-Level Credit Assignment

For negative-advantage rollouts, we apply a token-level mask to concentrate the penalty on tokens that are likely to correspond to the erroneous behavior. Given the student token log-probability $\log \pi_\theta(y_t)$ and normalized token entropy $\widehat{H}_t$, we define

$$s_t = -\log \pi_\theta(y_t) \left(1 - \widehat{H}_t\right). \tag{12}$$

The top $50\%$ of tokens according to $s_t$ are retained for negative-advantage updates. This preferentially retains tokens that were produced with relatively high confidence but low probability under the sampled distribution, while suppressing routine high-probability formatting tokens.

For a negative-advantage span, Eq. 11 is therefore replaced by

$$\mathcal{L}_{\mathrm{RL}}(y) = -A \frac{\sum_t m_t \log \pi_\theta(y_t \mid y_{<t}, x)}{\max(1, \sum_t m_t)}, \tag{13}$$

where $m_t \in \{0, 1\}$ is the token mask. Positive-advantage trajectories are reinforced over their complete scored span and are not masked.

### C.5 Privileged On-Policy Distillation

The chair produces two conceptually distinct outputs: a routing decision and an information-state insight. The routing decision is trained from the verifier-based reward, whereas the insight is trained using privileged on-policy distillation.

The teacher is the same base Qwen3-4B model as the student router, with the LoRA adapter disabled, but receives the complete set of decisive facts $z_{1:N}$ in its input. The student receives only its ordinary observation $x_t$. Thus,

$$\pi_T(\cdot \mid x_t, z_{1:N}) \quad \text{(privileged teacher)}, \tag{14}$$

$$\pi_c(\cdot \mid x_t) \quad \text{(student chair)}. \tag{15}$$

The decisive facts are available only during training and are never supplied to the student at inference time.

We distill only the textual insight rather than the teacher's routing decision. This is important because a privileged teacher can select an agent using information that the student cannot observe. Directly distilling such a routing decision would therefore provide supervision that is unavailable at inference time.

For a student-generated insight $h = (h_1, \ldots, h_L)$, we compute a per-token forward KL divergence,

$$D_t = D_{\mathrm{KL}}\left[\pi_T(\cdot \mid x_t, z_{1:N}, h_{<t}) \,\|\, \pi_c(\cdot \mid x_t, h_{<t})\right]. \tag{16}$$

The teacher scores the student's own generated completion, so the distillation is on-policy rather than on teacher-generated trajectories.

We further weight each token according to the student's uncertainty and its disagreement with the teacher. Let $\widehat{H}_t$ be min–max normalized student entropy and $\widehat{D}_t$ be min–max normalized teacher–student divergence. The token weight is

$$w_t = \widehat{H}_t + \widehat{D}_t - \widehat{H}_t \widehat{D}_t. \tag{17}$$

This is a soft-OR weighting: tokens are emphasized when the student is uncertain or disagrees with the privileged teacher, while tokens that are both confident and already aligned contribute little to the distillation objective.

The resulting insight loss is

$$\mathcal{L}_{\mathrm{OPD}} = \frac{1}{L} \sum_{t=1}^{L} w_t D_t, \tag{18}$$

with a per-token KL ceiling of 20. The KL temperature is 1.0.

Importantly, OPD is applied only to rollouts whose group-relative advantage is positive:

$$\mathcal{S}_+ = \{j : A_j > 0\}. \tag{19}$$

Thus, the privileged teacher provides a target for trajectories that the student policy has already produced relatively successfully, while unsuccessful trajectories are trained through the verifier-based reinforcement-learning signal instead.

The router's total optimization objective is

$$\mathcal{L}_c = \mathcal{L}_{\mathrm{RL}}^{\mathrm{route}} + \lambda_{\mathrm{OPD}} \mathcal{L}_{\mathrm{OPD}}, \tag{20}$$

where $\lambda_{\mathrm{OPD}} = 1.0$ in our experiments. The routing and insight losses update the same router LoRA parameters in a single optimizer step.

### C.6 Participant Training

All participants share a single policy $\pi_a$. Participant identity is provided through the prompt and private-information context rather than through separate model parameters. For a group of trajectories generated with the router frozen, participants receive the task reward

$$r_j^a = R_{\mathrm{train}}(\tau_j), \tag{21}$$

and the corresponding group-relative advantages are computed using Eq. 9.

The participant loss is

$$\mathcal{L}_a = -\frac{1}{|\mathcal{Y}_a|} \sum_{(j,y) \in \mathcal{Y}_a} A_j \frac{1}{|y|} \sum_{k=1}^{|y|} \log \pi_a(y_k \mid y_{<k}, x), \tag{22}$$

where $\mathcal{Y}_a$ contains the textual responses generated by participants, including the settlement generated by the decision maker. No privileged teacher signal is used for participant updates.

### C.7 Alternating Co-Evolution

Training alternates between router and participant updates. We use four phases over the 720 training scenarios:

$$\text{router} \rightarrow \text{participant} \rightarrow \text{router} \rightarrow \text{participant}. \tag{23}$$

Each phase contains 180 scenarios. Within a phase, the counterpart policy is frozen, and groups of four trajectories are sampled for every scenario. Eight scenarios are accumulated before each optimizer step, corresponding to $8 \times 4 = 32$ rollouts per optimization step.

Freezing the counterpart policy ensures that the trajectories within a group differ only through the policy being optimized. Consequently, the group-relative advantage provides credit to the active policy rather than conflating simultaneous changes in both policies. The alternating procedure also allows improvements to propagate between the two roles: improved routing exposes more useful private information, which changes the participant's learning problem, while improved disclosure makes future routing decisions more informative. Figure 4 shows the corresponding routing-precision dynamics across the four alternating phases for the different policy-gradient algorithms namely, RLOO Ahmadian et al. (2024), GRPO Shao et al. (2024), GSPO Zheng et al. (2025) and Dr GRPO Liu et al. (2025b).

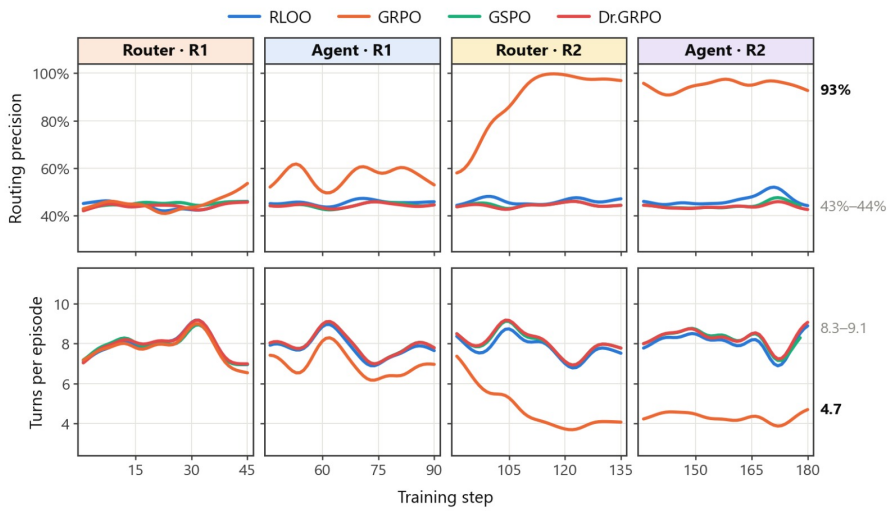



Figure 4: Routing precision of the router during training of Co-Adapt with four policy-gradient algorithms (RLOO, GRPO, GSPO, Dr.GRPO) over four alternating phases (Router–R1, Agent–R1, Router–R2, Agent–R2; 45 steps each). Values at the right edge are the final-step values.

### C.8 Model and Optimization Details

The router uses Qwen3-4B and the participant policy uses Qwen3-8B. Both models are trained using LoRA adapters in bfloat16 precision. The router uses LoRA rank $r = 32$, $\alpha = 64$, and dropout $0.05$, while the participant policy uses $r = 16$, $\alpha = 32$, and dropout $0.0$. LoRA targets are automatically discovered over the linear layers of the corresponding model.

During rollout generation, the router uses temperature $0.3$, top-$p = 0.95$, and top-$k = 20$, with a maximum of 24 tokens for routing and 320 tokens for insight generation. Participants use temperature $0.8$, top-$p = 0.95$, and top-$k = 20$, with maximum generation lengths of 160 tokens for ordinary responses and 400 tokens for settlements.

We optimize the router with AdamW at learning rate $2 \times 10^{-5}$ and the participant policy with AdamW at learning rate $10^{-5}$. The router learning rate uses a linear warm-up over the first 10 router optimizer steps. The global gradient norm is clipped to $1.0$. We do not use a PPO importance-ratio correction or clipping: each sampled trajectory is scored under the current policy in the single update pass, so the policy ratio is identically one.

All losses are accumulated over the scenarios assigned to an optimization step and then applied as one optimizer update. The training configuration therefore uses group-relative rewards for credit assignment, shaped verifier feedback for dense learning signals, privileged OPD for the chair's information-state representation, and alternating updates to couple routing and participant disclosure.

## D Additional Experimental Results

This appendix reports three further experiments with the setup of Section 5: additional router and participant model pairs, a domain-wise breakdown of the main result, and a seed robustness study. All runs use the training recipe and hyperparameters of Appendix C.8 and the same test split as Table 2. Metrics are those of Table 2.

### D.1 Additional Router and Participant Backbones

Table 6 reports four further pairs. Each changes one part of the architecture relative to the Qwen3 system of Table 2: a larger router with smaller participants (Qwen3 8B router, 4B participants), smaller participants with the same router (Qwen3 4B router, 4B participants), a router from a different family (Gemma 3 4B router, Qwen3 8B participants), and participants from a different family

| | Seen (n = 180) | | | | | Unseen (n = 200) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | **Pass Rate ↑** | **Solved ↑** | **Content ↑** | **Prov. ↑** | **Turns** | **Pass Rate ↑** | **Solved ↑** | **Content ↑** | **Prov. ↑** | **Turns** |
| *Qwen3: 8B router, 4B agents* | | | | | | | | | | |
| Vanilla | 0.77 | **0.206** | 0.74 | 0.94 | 5.96 | 0.73 | 0.150 | 0.71 | 0.85 | 5.10 |
| **Co-Adapt** | **0.79** | 0.183 | **0.75** | **0.98** | 4.08 | **0.77** | **0.185** | **0.74** | **0.92** | 4.18 |
| *Qwen3: 4B router, 4B agents* | | | | | | | | | | |
| Vanilla | 0.63 | 0.111 | 0.61 | 0.71 | 3.86 | 0.58 | 0.085 | 0.58 | 0.59 | 3.59 |
| **Co-Adapt** | **0.76** | **0.150** | **0.72** | **0.97** | 4.53 | **0.75** | **0.160** | **0.72** | **0.90** | 4.74 |
| *Gemma 3 4B router, Qwen3 8B agents* | | | | | | | | | | |
| Vanilla | 0.41 | 0.033 | 0.43 | 0.31 | 2.51 | 0.38 | 0.010 | 0.42 | 0.25 | 2.19 |
| **Co-Adapt** | **0.82** | **0.261** | **0.79** | **0.97** | 4.01 | **0.79** | **0.195** | **0.76** | **0.92** | 4.19 |
| *Qwen3 4B router, Llama 8B agents* | | | | | | | | | | |
| Vanilla | 0.67 | **0.122** | 0.64 | 0.79 | 5.72 | 0.63 | 0.085 | **0.62** | 0.70 | 6.57 |
| **Co-Adapt** | **0.69** | 0.061 | **0.65** | **0.88** | 5.00 | 0.63 | 0.085 | 0.61 | **0.79** | 5.67 |

Table 6: Results with additional router and participant backbones on the ELICITBENCH test set. Splits and metrics are as in Table 2. Vanilla uses the same router and participant models without training. Within each pair, the better of Vanilla and CO-ADAPT is shown in bold.

| | | Pass Rate ↑ | | Solved ↑ | | Content ↑ | | Prov. ↑ | | Turns | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Domain** | $n$ | Vanilla | Co-Adapt | Vanilla | Co-Adapt | Vanilla | Co-Adapt | Vanilla | Co-Adapt | Vanilla | Co-Adapt |
| *Seen domains* | | | | | | | | | | | |
| Defense | 20 | 0.71 | **0.78** | 0.300 | **0.350** | 0.65 | **0.71** | 0.83 | **1.00** | 3.85 | 4.30 |
| Education | 20 | 0.56 | **0.73** | 0.000 | **0.150** | 0.56 | **0.68** | 0.60 | **0.95** | 3.55 | 3.68 |
| Entertainment | 20 | 0.57 | **0.78** | 0.100 | **0.150** | 0.56 | **0.75** | 0.65 | **0.95** | 3.90 | 3.95 |
| Finance | 20 | 0.76 | **0.90** | 0.300 | **0.400** | 0.75 | **0.88** | 0.84 | **1.00** | 5.20 | 4.65 |
| Healthcare | 20 | 0.64 | **0.79** | 0.100 | 0.100 | 0.62 | **0.74** | 0.68 | **1.00** | 3.10 | 3.90 |
| Legal | 20 | 0.75 | **0.81** | 0.150 | 0.150 | 0.72 | **0.79** | 0.86 | **0.90** | 4.30 | 4.20 |
| Manufacturing | 20 | 0.73 | **0.85** | 0.100 | **0.250** | 0.72 | **0.82** | 0.85 | **1.00** | 4.00 | 4.35 |
| Software & Technology | 20 | 0.58 | **0.80** | 0.050 | **0.150** | 0.58 | **0.76** | 0.52 | **0.93** | 3.50 | 4.20 |
| Workplace Interpersonal | 20 | 0.58 | **0.90** | 0.200 | **0.550** | 0.65 | **0.85** | 0.41 | **1.00** | 3.30 | 4.00 |
| All seen domains | 180 | 0.65 | **0.81** | 0.144 | **0.250** | 0.65 | **0.78** | 0.67 | **0.98** | 3.86 | 4.14 |
| *Unseen domains* | | | | | | | | | | | |
| Friends, Family & Informal | 100 | 0.48 | **0.77** | 0.030 | **0.160** | 0.47 | **0.73** | 0.48 | **0.97** | 2.67 | 4.15 |
| Commerce / Bargaining | 100 | 0.65 | **0.79** | 0.190 | **0.260** | 0.67 | **0.78** | 0.56 | **0.82** | 3.22 | 4.43 |
| All unseen domains | 200 | 0.57 | **0.78** | 0.110 | **0.210** | 0.58 | **0.75** | 0.53 | **0.89** | 2.94 | 4.29 |
| **All domains** | 380 | 0.61 | **0.80** | 0.126 | **0.229** | 0.61 | **0.76** | 0.59 | **0.93** | 3.38 | 4.22 |

Table 7: Domain-wise results of Vanilla and CO-ADAPT with the Qwen3 4B router and Qwen3 8B participants on the ELICITBENCH test set. Pass Rate and Solved are defined as in Table 2. Content and Prov. are the fractions of content and provenance checks passed per scenario, computed against each domain's own number of checks, and Turns is the mean number of turns to settlement over settled episodes. Within each domain, the better of Vanilla and CO-ADAPT is shown in bold.

(Qwen3 4B router, Llama 8B participants). Together with the three pairs in Table 2, this gives seven pairs. We describe the trends across them rather than the individual numbers.

**Router size matters before training and much less after it.** Without training, the 8B router is far ahead of the 4B router with the same 4B participants. After training the two are close. Training a small router therefore recovers most of what a larger router gives out of the box. The same pattern appears with the Phi system in Table 2: pairs whose untrained router is already strong gain little, pairs whose untrained router is weak gain a lot, and the trained systems end in a narrow band of Pass Rate regardless of where they started.

**Participant size shows up in the settlement, not in elicitation.** With the same 4B router, the 4B and 8B participants reveal the same facts after training and reach similar Provenance. The difference between them is in Content and Solved: the larger participants turn the revealed facts into a correct settlement more often. Participant capacity therefore matters for the decision maker's final step, once elicitation has been learned.

| Method | Seed | Pass Rate ↑ | Solved ↑ | Content ↑ | Prov. ↑ | Turns |
|---|---|---|---|---|---|---|
| *Seen* (n = 180) | | | | | | |
| Vanilla | 42 | 0.653 | 0.144 | 0.649 | 0.672 | 3.86 |
| | 123 | 0.646 | 0.144 | 0.636 | 0.680 | 3.84 |
| | 2024 | 0.651 | 0.139 | 0.634 | 0.730 | 3.89 |
| | mean ± std | 0.650 ± 0.004 | 0.143 ± 0.003 | 0.640 ± 0.008 | 0.694 ± 0.031 | 3.86 ± 0.03 |
| Co-Adapt | 42 | 0.813 | 0.250 | 0.778 | 0.974 | 4.14 |
| | 123 | 0.808 | 0.256 | 0.771 | 0.962 | 4.07 |
| | 2024 | 0.807 | 0.222 | 0.773 | 0.978 | 4.05 |
| | mean ± std | 0.809 ± 0.003 | 0.243 ± 0.018 | 0.774 ± 0.004 | 0.972 ± 0.009 | 4.09 ± 0.05 |
| *Unseen* (n = 200) | | | | | | |
| Vanilla | 42 | 0.566 | 0.110 | 0.575 | 0.524 | 2.94 |
| | 123 | 0.582 | 0.120 | 0.591 | 0.566 | 3.05 |
| | 2024 | 0.573 | 0.090 | 0.591 | 0.538 | 3.03 |
| | mean ± std | 0.574 ± 0.008 | 0.107 ± 0.015 | 0.586 ± 0.009 | 0.543 ± 0.021 | 3.01 ± 0.05 |
| Co-Adapt | 42 | 0.779 | 0.210 | 0.752 | 0.882 | 4.29 |
| | 123 | 0.773 | 0.215 | 0.749 | 0.892 | 4.08 |
| | 2024 | 0.801 | 0.220 | 0.775 | 0.924 | 4.09 |
| | mean ± std | 0.784 ± 0.015 | 0.215 ± 0.005 | 0.759 ± 0.015 | 0.899 ± 0.022 | 4.15 ± 0.12 |
| *All* (n = 380) | | | | | | |
| Vanilla | 42 | 0.608 | 0.126 | 0.610 | 0.591 | 3.38 |
| | 123 | 0.612 | 0.132 | 0.612 | 0.618 | 3.42 |
| | 2024 | 0.610 | 0.113 | 0.611 | 0.626 | 3.44 |
| | mean ± std | 0.610 ± 0.002 | 0.124 ± 0.009 | 0.611 ± 0.001 | 0.612 ± 0.018 | 3.41 ± 0.03 |
| Co-Adapt | 42 | 0.795 | 0.229 | 0.764 | 0.924 | 4.22 |
| | 123 | 0.789 | 0.234 | 0.759 | 0.924 | 4.07 |
| | 2024 | 0.804 | 0.221 | 0.774 | 0.949 | 4.07 |
| | mean ± std | 0.796 ± 0.007 | 0.228 ± 0.007 | 0.766 ± 0.008 | 0.932 ± 0.014 | 4.12 ± 0.08 |

Table 8: Seed robustness of CO-ADAPT with the Qwen3 4B router and Qwen3 8B participants on the ELICITBENCH test set. Each seed is an independent training run followed by evaluation on the same test scenarios; seed 42 is the run reported in Table 2. Metrics are as in Table 2, shown to three decimals. Shaded rows give the mean and sample standard deviation over the three seeds.

**The router family can be changed; the participant family is the sensitive part.** The two mixed pairs behave differently. A Gemma 3 router with Qwen3 participants is the weakest untrained system in the paper: the router hands the floor to the decision maker after two or three turns, before the facts are on the record. After training it is the strongest pair. The routing policy is learned from the verifier and does not depend on the router sharing a family with the participants. With Llama participants and a Qwen3 router, the opposite happens. Training improves elicitation, so Provenance rises and the participants reveal all of their decisive facts, but Content and Solved stay where they were. The participant policy is the part that has to learn the settlement, and the Llama participants did not learn it under our recipe.

**Training normalises the length of the interaction.** Untrained systems sit at one of two extremes. Some settle too early, before the facts are revealed (the Gemma 3 router, and the Qwen3 4B router on the unseen domains). Others keep the meeting going well past the point where all facts are on the record (the Qwen3 8B router, the Phi system, and the Llama participants). After training, every pair settles in about four turns, which is close to the minimum needed to reveal each decisive fact once and then settle. Training shortens long meetings and lengthens short ones.

**Elicitation gains are uniform across pairs; settlement gains are not.** For every pair, training raises Provenance to a high level and brings the number of revealed decisive facts to its maximum. What differs between pairs is whether the revealed facts produce a passing settlement. The size of

the gain in Pass Rate is therefore decided mainly by the participant model, and the remaining errors of the trained systems are settlement errors rather than missing facts.

**Transfer to unseen domains holds for every trained pair.** Without training, the gap between Seen and Unseen varies from pair to pair. After training it is small for every pair except the Llama one. The routing behaviour learned on the nine training domains carries over to the two held-out domains for each backbone.

In short, CO-ADAPT makes the routing side of the system largely independent of the router's size and family, and moves the remaining differences between architectures to the participants' ability to write a correct settlement.

### D.2 DOMAIN-WISE RESULTS

Table 7 breaks the Qwen3 result of Table 2 down by domain. CO-ADAPT improves Pass Rate, Content and Provenance in every domain. Solved improves in nine domains and is unchanged in Healthcare (0.100) and Legal (0.150).

The largest gain is in Workplace Interpersonal, where Pass Rate rises from 0.58 to 0.90, Solved from 0.200 to 0.550, and the number of provenance checks passed from 0.90 to 2.20. This domain has the most provenance checks per scenario (1.90, Table 5), so its settlements depend most on facts being revealed and cited. Software & Technology (0.58 to 0.80) and Entertainment (0.57 to 0.78) follow. The smallest gains are in Legal (0.75 to 0.81) and Defense (0.71 to 0.78), where Vanilla is already strong.

On the two unseen domains, the gain is larger in Friends, Family & Informal (0.48 to 0.77) than in Commerce / Bargaining (0.65 to 0.79). Friends, Family & Informal is the domain where Vanilla is weakest. Its untrained system settles after 2.67 turns on average, before most facts are revealed, and the trained system takes 4.15 turns.

Turns rise in nine of the eleven domains and fall in Finance (5.20 to 4.65) and Legal (4.30 to 4.20). In most domains the untrained decision maker settles early, and the trained router asks more participants before settling. Where Vanilla already takes many turns, as in Finance, training makes the interaction shorter.

### D.3 SEED ROBUSTNESS

Table 8 repeats the Qwen3 experiment with three seeds (42, 123 and 2024). Each seed is a separate training run followed by evaluation on the test set; seed 42 is the run reported in Table 2.

On every seed and every split, CO-ADAPT is above Vanilla on Pass Rate, Solved, Content and Provenance. The spread across seeds is small. The standard deviation of Pass Rate is at most 0.015 on any split, while the gain from training is 0.16 on Seen and 0.21 on Unseen, at least ten times the spread. Solved on Seen varies the most across seeds (0.222 to 0.256); with 180 scenarios, one scenario changes it by 0.006. Turns on Unseen vary between 4.08 and 4.29.

The number of decisive facts revealed per episode is the same for all three seeds: 3.00 on Seen and 3.02 on Unseen, with zero standard deviation, against 2.16 and 1.71 for Vanilla. The trained participants therefore reveal all of their decisive facts in every episode. The remaining gap between the trained Pass Rate (0.81 and 0.78) and 1.0 comes from the settlement, not from unrevealed facts.

## E HUMAN EVALUATION

While the programmatic evaluation provides deterministic measures of settlement correctness and information provenance, these metrics do not directly capture the quality of the interaction itself. We therefore conduct a human evaluation on the held-out test set to assess whether the behavior exhibited by the agents corresponds to the information-elicitation process targeted by CO-ADAPT.

### E.1 Evaluation Dimensions

We evaluate each agent trajectory along three dimensions that correspond to the objectives of CO-ADAPT: routing quality, information disclosure, and decision quality. The dimensions are defined to be observable from the scenario context and the resulting multi-agent interaction transcript.

- **Routing Quality.** The extent to which the chair directs questions toward participants who are relevant for obtaining information needed for the decision.
- **Information Disclosure.** The extent to which participants appropriately disclose decision-relevant private information when it is needed, while avoiding unnecessary disclosure of unrelated private information.
- **Decision Quality.** The extent to which the final decision correctly reflects the relevant information and satisfies the scenario's decision requirements.

Each trajectory is rated independently on a five-point ordinal scale. The corresponding questions presented to the evaluators are:

- **Routing Quality:** "Did the chair direct questions toward the participant(s) most relevant for obtaining the information needed for the decision?"
- **Information Disclosure:** "Did participants appropriately disclose relevant private information when it was needed, without unnecessary disclosure of unrelated private information?"
- **Decision Quality:** "Does the final decision correctly reflect the relevant information and satisfy the scenario's decision requirements?"

For the information-disclosure dimension, evaluators consider both the relevance and timing of disclosures. A high score therefore requires participants to disclose information that is useful for the decision when it becomes relevant, rather than indiscriminately revealing private information.

### E.2 Human Evaluation Protocol

We conduct the human evaluation on 100 held-out scenario from test set. Three postgraduate evaluators independently assess the selected agent trajectories. Evaluators are presented with the scenario context and the complete multi-agent interaction transcript, but are not provided with the model identity or the corresponding programmatic evaluation scores. Each trajectory is rated independently on the four evaluation dimensions described above using a five-point ordinal scale.

The evaluators are instructed to assess the interaction based only on the information available in the scenario and the observed transcript. In particular, evaluators distinguish whether the chair directed the interaction toward relevant participants (routing quality), and whether participants disclosed relevant private information appropriately and without unnecessary disclosure (information disclosure).

The evaluators are compensated for their time according to the university's standard compensation norms. For each dimension, the final human score for a trajectory is computed as the mean rating across the three independent evaluators. We additionally measure inter-rater agreement to assess the reliability and consistency of the human judgments.

The evaluation is designed to complement, rather than replace, the programmatic evaluation. In particular, the human judgments provide an independent assessment of the interaction behaviors targeted by CO-ADAPT, while the automatic metrics provide deterministic measurements of settlement correctness and information provenance.

### E.3 Human Evaluation Results

Table 9 reports the human evaluation results. Each dimension is scored on a five-point scale, with higher values indicating better performance.

### E.4 Inter-Rater Reliability

To assess the consistency of the human judgments, we measure inter-rater agreement separately for each evaluation dimension. We use Krippendorff's $\alpha$ with an ordinal distance function, since each

| **Method** | Route Q. ↑ | Inf. Q. ↑ | Dec. Q. ↑ |
|---|---|---|---|
| Vanilla | 2.7 | 2.9 | 3.0 |
| **Co-Adapt** | **4.2** | **4.3** | **4.1** |
| Gemma Vanilla | 2.8 | 2.9 | 2.8 |
| **Gemma Co-Adapt** | **4.0** | **3.9** | **3.7** |

Table 9: Human evaluation on the held-out test set. All dimensions are rated on a 1–5 scale, where higher is better.

| **Dimension** | **Krippendorff's $\alpha$** |
|---|---|
| Routing Quality | 0.76 |
| Information Disclosure | 0.84 |
| Decision Quality | 0.80 |
| Overall | 0.81 |

Table 10: Inter-rater reliability for the human evaluation. Krippendorff's $\alpha$ is computed using an ordinal distance function.

dimension is evaluated using a five-point ordinal scale and multiple independent evaluators rate the same trajectories.

# F PROMPTS

This section reproduces the four prompts used by the scenario-generation pipeline. Curly-brace placeholders (e.g. `{facts_block}`) mark where scenario-specific content is substituted at call time.

**Prompt 1: Challenger Scenario-Generation**

```
Write ONE scenario at a time:  a group decision where the information needed to decide
correctly is split across the group; no single person has all of it.  Output only the
scenario object, never a conversation, transcript, or grade.

Certificate it must earn:  ALONE (decision-maker sees only shared facts + own private
facts, no discussion, decides immediately) -> usually WRONG. TOGETHER (everyone sees
shared facts + own private facts + everything said so far, and can talk) -> usually RIGHT.
If either half is false the scenario is worthless:  one solvable alone teaches nothing
about coordination, one the group also can't solve teaches nothing at all.

Structural rules:  3, 4, or 5 agents, with logic that does not depend on the exact
count; exactly one decision-maker; every other agent owns at least one decisive fact,
no exceptions or bystanders (checked mechanically); zero hints that information is hidden
or split:  every shared_context sentence must read as something a person in the room could
actually observe; no agent's private facts ever appear in another agent's view; a shared
fact must give everyone a normal, ordinary reason to speak (a habit, a policy, a routine
check-in) before the decision is final, never a hint that something is being withheld.

Decisive facts:  each private fact must change something concrete in the final decision
(a number, an assignment, a date, a commitment), or it does not belong and should be cut.
For every decisive fact, record fact_id, owner, exactly which check IDs it flips from fail
to pass, and a one-sentence "why" naming the exact number or choice that changes (never
"this fact is important").  The decision-maker must not be able to pass by being generous,
fair, or by guessing a stereotype; make private facts specific enough that only genuine
use of them passes.

Checks:  every acceptance condition becomes a single boolean Python expression,
never a sentence:  a content check (verifiable from the settlement JSON alone) or a
provenance check (also requires the fact was spoken aloud by its owner AND cited by the
decision-maker).  Minimum 4 content checks, 1 provenance check; fewer is an automatic
rejection.  Never soften a threshold implied by a private fact (rounding down, widening a
range, "at least half" -> "some"); this is the single most common way a scenario silently
becomes solvable alone.

Five-name law:  every check may reference ONLY decisions, credited_facts, commitments,
justification_fact_ids, revealed, plus a comprehension's own loop variable,
using literals, ==/!=/</<=/>/>=, in/not in, and/or/not, any/all/len/sum, and
.values/.keys/.items/.get.  No if/else ternaries, imports, lambdas, or any other name
or method.  Every check must be a single expression evaluating to True or False.

Self-check before returning:  solvable (>=1 settlement passes every check); failable
(>=1 plausible settlement fails a check); not trivial (a generic fair/equal/cautious
settlement does not pass everything); faithful (no threshold was softened); no leaks
in shared_context; decisive facts complete (one per non-decision-maker minimum, each
with a specific why); enough checks; valid Python throughout.  Output JSON only,
matching:  scenario_id, scenario_type, description, agents, shared_context, private_facts,
views, decision_maker, interaction_config, settlement_schema, acceptance_conditions,
content_checks, provenance_checks, decisive_facts.

On rejection you get a tag, diagnosis, evidence, and fix instructions plus your prior
scenario JSON; revise the same scenario, change only what caused the rejection, and
resubmit the full object as JSON only.  MALFORMED = a check is broken or shared_context
leaked the setup -> rewrite the leaking sentence or the check.  LEAKED = the lone
decision-maker passed too often, the hidden information wasn't truly required ->
tighten every loose check back to exactly what the private fact states.  UNCOORDINATED
= the information gap is real but the group still failed too often -> strengthen the
consultation norm in shared_context so every fact-holder gets a real, unmissable moment to
speak; never fix this by making an agent more talkative or assertive; fix the environment,
never the agents.
```

**Prompt 2: Verifier**

```
You get one candidate scenario JSON and check it before any rollouts run:  a single cheap
call, reasoning about the scenario object alone, not running the weak arm or strong arm.

(1) Leakage:  look only at the decision-maker's view (shared_context plus their own
entries in private_facts).  Could a competent agent pass every content_check using only
this view, by stating things, by an obvious inference, or by a generic trope (fairness,
seniority, "just ask everyone")?  If yes, that's a leak.  (2) Satisfiable:  build one
hypothetical settlement, using facts from anywhere in the scenario as if every private
```

fact had been revealed and used, that passes every content_check and provenance_check. If none exists, the checks are broken. (3) Falsifiable: build one plausible settlement (what a reasonable decision-maker might produce while missing some hidden facts) that fails at least one check. If every plausible settlement passes everything, the checks are trivial; also confirm no check is automatically guaranteed by another (redundant or vacuous).

Also confirm: no sentence in shared_context describes the scenario's own structure or hints information is hidden or missing: shared facts must read like ordinary things a person in the room would know; every fact ID used anywhere actually exists in shared_context or private_facts; every entry in views is exactly that agent's shared facts plus only their own private facts, with no other agent's private fact IDs leaking in; every check in content_checks and provenance_checks is one Python expression using only the five allowed names, every string literal quoted, evaluating to True or False.

Return only this JSON: verdict (PASS/REJECT), tag (MALFORMED or null), checks (leakage_free, satisfiable, falsifiable, fact_ids_valid, views_valid, checks_are_valid_python, all booleans), diagnosis (plain sentences naming exactly what's wrong, or "no issues found"), evidence (the specific sentence, check id, or fact id that's the problem), fix_instructions (one concrete instruction for the Challenger's revision, or null if PASS). You are the only stage that produces MALFORMED; LEAKED and UNCOORDINATED come later, from the weak-arm and strong-arm rollout gates.

### Prompt 3: Lone Agent Arm

You are {decision_maker_name}, the {decision_maker_role}. You must make a final, binding decision right now, completely alone. There is no discussion, no committee, no one else in the room, and no opportunity to ask questions. This is the ONLY information you have or will ever have: {facts_block}

Strict rules: base your decision ONLY on the facts listed above; do not assume, infer, invent, or guess any fact, number, motive, or circumstance that is not explicitly stated. Do not default to a generic fair split, an equal division, or a stereotype about what "usually" happens in situations like this; decide strictly from the facts given. You must output a single JSON object and nothing else: no prose, no markdown fences, no explanation before or after it. The JSON must match this exact shape, every field filled with a real value, never a placeholder: {schema_block}

Decide now and output ONLY the JSON object.

### Prompt 4: Society of Agents Arm

You are {name}, the {role}, in a live group conversation with other people. What everyone already knows: {shared_block} What ONLY you privately know, which nobody else knows unless you tell them: {private_block} The conversation so far, in order: {history_block}

Strict rules, no exceptions: a private fact only counts as disclosed if you use the 'reveal' action and name its exact fact_id: mentioning its content inside a plain 'say' does NOT count and will NOT be credited to you. Never assume, reference, or act on information you were not explicitly given above or explicitly told by another speaker earlier in this conversation. Take exactly ONE action this turn, never combine actions. If you are the decision-maker settling, cite ONLY facts that were your own private facts, shared facts, or facts another agent explicitly revealed earlier, in justification_fact_ids; never a fact you were never told.

Available actions: say {"type":"say","text":"..."}; reveal {"type":"reveal","fact_id":"<a fact_id you own and have not yet revealed>","text":"..."}; settle (decision-maker only, once allowed) {"type":"settle","settlement":<object matching this shape>}: {schema_block}

Respond with ONLY a single JSON object representing your one action: no prose, no markdown, no explanation.

```
{
  "scenario_type": "sensitive_meeting_facilitation",
  "description": "A team lead must finalize who will facilitate a sensitive retrospective
     after a difficult client launch and set the meeting's safeguards and schedule.",
  "domain": "workplace_interpersonal",
  "agents": [
    {"agent_id": "A1", "name": "Imani", "role": "Team Lead"},
    {"agent_id": "A2", "name": "Priya", "role": "Teammate"},
    {"agent_id": "A3", "name": "Theo", "role": "Project Coordinator"}
  ],
  "decision_maker": "A1",
  "shared_context": {
    "S1": "The team is holding a retrospective about a difficult client launch involving
     Mara, Theo, and the client sponsor.",
    "S2": "Imani makes the final decision about the facilitator, meeting safeguards, and
     schedule.",
    "S3": "Mara has been proposed as a possible facilitator because she coordinated much of
     the launch.",
    "S4": "Mara is unavailable for this consultation, so Imani's normal practice is to ask
     the participating colleagues, Priya and Theo, for relevant input and then finalize the
     facilitator and schedule without waiting for Mara."
  },
  "private_facts": {
    "PF1": {"owner": "A2",
            "text": "Priya observed an unresolved disagreement between Mara and the client
     sponsor during an earlier project, and the sponsor agreed to participate in this
     retrospective only if Theo facilitates and Mara is not the sole facilitator. Priya did
     not raise this with Imani because it came up in a private debrief that Imani did not
     attend."},
    "PF2": {"owner": "A3",
            "text": "Theo has a school-pickup commitment on Thursday. Thursday from 15:30
     through 16:15 is the only time Theo can attend the retrospective on any day; Theo
     cannot attend at any other time or on any other day. Theo told the project coordinator'
     s scheduling inbox, but that availability has not yet been consolidated for Imani."}
  },
  "views": {
    "A1": ["S1", "S2", "S3", "S4"],
    "A2": ["S1", "S2", "S3", "S4", "PF1"],
    "A3": ["S1", "S2", "S3", "S4", "PF2"]
  },
  "decisive_facts": [
    {"fact_id": "PF1", "owner": "A2", "flips": ["C1", "C2", "C6", "P1"],
     "why": "The sponsor's prior conflict and participation condition change the facilitator
      choice from proposed Mara to Theo, change sole facilitation from allowed to false, and
      require Priya's fact to be credited."},
    {"fact_id": "PF2", "owner": "A3", "flips": ["C3", "C4", "C6", "P1"],
     "why": "Theo's exact only available slot changes the meeting schedule to Thursday
     15:30-16:15 rather than any other day, time, or an end time later than 16:15."}
  ],
  "interaction_config": {"turn_order": ["A2", "A3", "A1"], "turn_cap": 10},
  "settlement_schema": {
    "decisions": {"facilitator": "string", "mara_is_sole_facilitator": "boolean", "
     meeting_day": "string", "meeting_start": "string", "meeting_end": "string"},
    "credited_facts": ["fact_id"],
    "commitments": [{"type": "string", "target": "string", "detail": "string"}],
    "justification_fact_ids": ["fact_id"]
  },
  "acceptance_conditions": [
    "Theo must facilitate the retrospective.",
    "Mara must not be the sole facilitator.",
    "The retrospective must occur on Thursday from 15:30 through 16:15, the only time Theo
     can attend on any day.",
    "Both relevant private facts must be credited and cited by Imani after being spoken
     aloud."
  ],
  "content_checks": {
    "C1": "decisions['facilitator'] == 'Theo'",
    "C2": "decisions['mara_is_sole_facilitator'] == False",
    "C3": "decisions['meeting_day'] == 'Thursday' and decisions['meeting_start'] ==
     '15:30'",
    "C4": "decisions['meeting_end'] == '16:15'",
    "C6": "'PF1' in credited_facts and 'PF2' in credited_facts"
  },
  "provenance_checks": {
    "P1": "'PF1' in revealed and 'PF1' in justification_fact_ids and 'PF2' in revealed and '
     PF2' in justification_fact_ids"
  }
}
```

Listing 1: A complete scenario from the Workplace Interpersonal domain.

| Domain | Setting | Decision-maker roles (examples) | Decision types (examples) |
|---|---|---|---|
| Healthcare | Hospitals, clinics, care networks | Charge nurse, attending physician, case manager, pharmacy director, nursing supervisor | Bed assignment, discharge timing, operating-room block time, formulary approval, staffing, referral prioritization |
| Software & Technology | Product and engineering organizations | Engineering manager, tech lead, product director, platform lead, security engineering manager | Release scope, incident post-mortems, vendor selection, migration go/no-go, on-call rotations, patch and feature-flag rollout |
| Defense | Military logistics units; routine planning only | Convoy commander, battalion executive officer, company commander, S-3 and S-4 officers, FOB commander | Route selection, equipment and radio allocation, fuel distribution, maintenance prioritization, training scheduling |
| Finance | Banks, funds, corporate finance teams | Credit officer, portfolio manager, finance director, treasury director, underwriting manager | Loan approval, portfolio rebalancing, budget sign-off, covenant waivers, credit-line renewal, hedging |
| Education | Schools and districts | Principal, assistant principal, department head, counselor, special-education coordinator | Student placement, behavior plans, field-trip logistics, IEP/504 accommodations, substitute coverage, room scheduling |
| Entertainment | Productions, festivals, studios, venues | Festival programmer, producer, showrunner, casting director, talent booker, venue operations director | Lineup scheduling, reshoot budgets, casting, rights clearance, live-broadcast go/no-go, venue capacity |
| Legal | Law firms and in-house legal teams | Lead attorney, general counsel, litigation director, chief legal officer | Settlement authority, contract renewal, litigation strategy, IP licensing, employment disputes, regulatory filings |
| Manufacturing | Factory floors and supply chains | Plant manager, procurement lead, operations director, quality director, safety officer | Line reallocation, supplier selection, maintenance scheduling, safety-incident response, shift staffing |
| Workplace Interpersonal | Interpersonal decisions at work | Manager, department head, HR business partner, team lead | Crediting disputes, promotions, conflict mediation, performance-improvement decisions, workload rebalancing |
| Friends, Family & Informal† | Personal decisions among friends or family; no institutional authority | Trip organizer, leaseholder, coordinating sibling, holder of a shared or inherited item | Trip planning, shared living, splitting a cost or an inherited item, coordinating care, allocating something scarce |
| Informal Commerce / Bargaining† | Informal price or terms negotiations; no institutional authority | Buyer, client, seller weighing an offer, sublet or swap arranger | Used-goods sale, freelance gig, service swap or barter, sublet price, allocating something several people want |

Table 11: The eleven domains of ELICITBENCH. Each domain contains 100 certified scenarios. Roles and decision types are examples from the domain specifications. †Held out entirely (unseen split); the other nine domains provide the training set and the seen split.