Mental-Models for Multi-Agent Systems

Hanan Gani
hgani@ucsd.edu

Lulu Shao
sishao@ucsd.edu

Manmohan Chandraker
mkchandraker@ucsd.edu

University of California, San Diego
La Jolla, CA, 92093

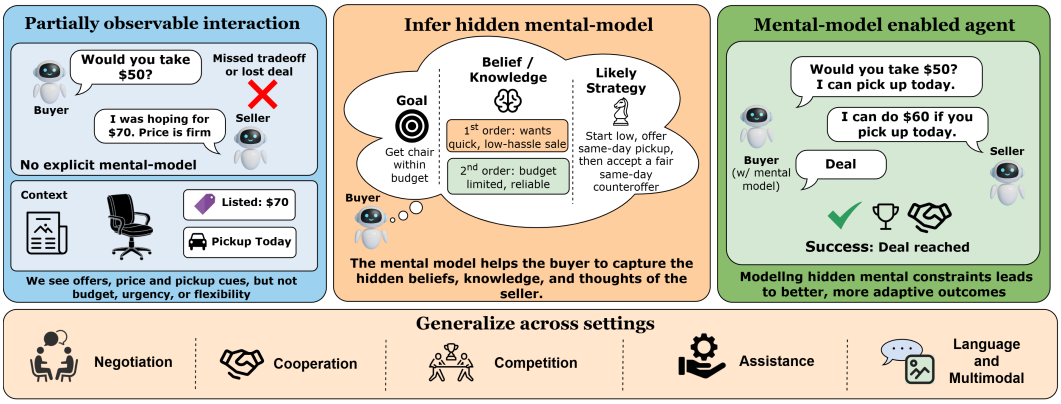


Figure 1: **Mental-model-enabled decision making under partial observability.** Observed behavior only partially reveals another agent's hidden goals, beliefs, constraints, and likely strategy. By inferring these latent mental states, a mental-model-enabled agent can choose more adaptive actions, improving outcomes while preserving cooperation across multi-agent settings.

Abstract

Large foundation models have accelerated progress toward general-purpose agents that interact with humans and other agents through language and multimodal signals. However, robust multi-agent decision-making requires reasoning about what other agents know, intend, and are likely to do under partial observability. Current agentic systems often operate through prompt design, memory, or end-to-end behavioral shaping, but typically do not learn an explicit partner-state representation that can be reused as a decision variable across tasks. We introduce *mental-model-enabled agents*, a framework that equips an agent with a latent mental model of its counterpart, allowing it to infer hidden beliefs, intentions, and likely reactions from the observed history and use these inferences to guide action selection. Our method learns an amortized recursive Theory-of-Mind representation, with first- and second-order mental-state structure, jointly with a belief-conditioned reward model that evaluates candidate actions relative to the inferred partner state. A policy is then learned under this belief-aware signal, yielding an agent that can act independently at inference time while retaining the benefits of explicit partner modeling. We evaluate the same framework on both language-only and multimodal benchmarks. Across these settings, explicit mental-state modeling consistently improves interaction quality and Theory-of-Mind performance over base agentic systems, showing that structured partner modeling is a useful inductive bias for

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

general multi-agent systems. Our code is publicly available at https://github.com/hananshafi/Mental-Models

1 Introduction

Modern AI, robotics, and cognitive science increasingly converge on a shared objective: building agents that can reason, plan, and act effectively in the world. The core idea is that intelligent systems, both biological and artificial, must construct "mental models" of the world. This is in line with Kenneth Craik's seminal work *The Nature of Explanation*, which proposed that organisms maintain small-scale internal representations of external reality [3]. Such mental models allow them to simulate possible actions, anticipate outcomes, and adapt effectively to changing environments. However, in interactive environments, the world is partly constituted by other agents whose goals, beliefs, and strategies are not directly observable. As a result, successful action requires modeling not only the environment, but also other minds, a capacity commonly studied as Theory of Mind (ToM) [8].

In humans, ToM is mediated by compact representations of a partner's latent mental states that support prediction and planning [8]. This perspective has motivated computational approaches that learn agent models directly from behavioral traces. ToMnet [22], for instance, showed that an observer can infer latent agent properties from past trajectories and use them to predict future behavior, demonstrating the value of explicit uncertainty over unobserved mental state and amortized inference that adapts rapidly to novel agents. These properties are particularly valuable in multi-agent settings, where partial observability is fundamental.

Large foundation models have accelerated progress toward general-purpose agents that interact with humans and other agents through language and multimodal signals. Yet robust interactive competence remains difficult. An agent must infer what other agents know, want, and are likely to do, and choose actions that improve outcomes under partial observability (Fig.1). Depending on the setting, this may involve cooperation, negotiation, assistance, competition, or adherence to conventions and norms. Empirically, despite strong conversational ability, current LLMs lag behind humans on core ingredients of interactive reasoning, including Theory-of-Mind-style inference [23, 29, 26], adherence to interaction norms [31], and robust performance across goal-driven multi-agent scenarios [36]. Humans succeed in such settings because they rapidly form and update beliefs about others from limited evidence [13, 28]. By contrast, current LLM agents typically do not maintain an explicit model of a partner's hidden goals, beliefs, or strategies. As a result, they may misread intent, commit to brittle plans, and struggle to adapt when the interaction unfolds in unexpected ways.

Most prior work on LLM-based multi-agent systems improves behavior through shaping or scaffolding rather than explicit partner-state estimation. Existing approaches include instruction tuning and preference-based reinforcement learning that steer generation [19, 25, 30], prompt-level mechanisms such as self-critique, reflection, and memory that condition the agent on textual summaries of prior interaction [1, 20], and multi-turn synthetic self-play that induces complex interaction skills such as consensus-building [18]. Across these approaches, partner dynamics are represented only implicitly either in the prompt text or trajectory summaries and not embedded in policy's latent dynamics. The result is the absence of an explicit belief state with clear semantics and quantified uncertainty, which limits interpretability, complicates reuse across tasks, and prevents partner modeling from being integrated systematically into different policy learners.

A long line of work in sequential decision-making under partial observability suggests that effective multi-agent behavior depends on maintaining beliefs over hidden state, including other agents' objectives, information states, and internal models [12, 5, 6]. Extending this principle to foundation-model agents is nontrivial. When beliefs about a partner are expressed only as short natural-language hypotheses or serialized summaries appended to prompts, they remain limited in expressivity, and downstream action selection becomes tightly coupled to prompt formulation. This motivates the central question of our work: *can an agent learn an amortized latent belief state from interaction history and use it as a stable training signal for multi-agent policy learning?*

In this paper, we address this question with *mental-model-enabled agents* for multi-agent systems under partial observability. Our approach learns a compact latent belief state from interaction history that captures hidden variables of other agents, including goals, beliefs, knowledge, strategies, and higher-order expectations. We then learn a reward model that evaluates candidate actions relative to this inferred state, rather than only the observable context. The resulting belief-conditioned reward is

used to train a policy under the learned mental model, while the final policy remains standalone at inference time. Together, these components define a general framework for ToM-guided decision making in multi-agent systems across diverse settings.

To validate the generality of our method, we evaluate it on both language-only and multimodal benchmarks for multi-agent systems. For language-only settings, we consider SOTOPIA [36], CRAIGSLIST-BARGAIN [9], TOMI [15], and BIGTOM [7]; for multimodal evaluation, we use MMROLE [4]. We also convert existing resources into ToM-supervised training data by augmenting them with mental-state annotations; Appendix Table 10 summarizes the additional supervision relative to the original releases. Across these benchmarks, explicit mental-state modeling consistently improves downstream performance. For example, on SOTOPIA, our approach with Qwen2.5-7B achieves an average relative gain of 11.5% over the base model across two partner settings, while on BIGTOM it improves paired TB/FB accuracy by 26.7 pp on average across the six reported settings.

In summary, our contributions are:

- We propose a general framework for ToM-guided multi-agent systems under partial observability, in which an agent learns an explicit latent belief state over other agents from observed history and uses it to support downstream decision making.
- We introduce a recursive mental-state architecture and a coupled mental-state/reward learning objective that make first- and second-order partner representations directly useful for evaluating candidate actions.
- We show how the learned mental signal can be distilled into standalone policies, so that explicit partner modeling improves training while deployment remains efficient.
- We evaluate the framework across language-only and multimodal benchmarks, and augment these resources with additional ToM supervision, utility signals, rationales, and hard negatives to support training and analysis.

2 Related Work

Multi-agent systems. Recent work has explored LLM-based multi-agent systems in which multiple agents coordinate through role specialization, dialogue, memory, and tool use. CAMEL studies cooperative role-conditioned interactions in an agent society setting [16]. MetaGPT, ChatDev, and AutoGen emphasize structured workflows, standardized operating procedures, and programmable conversation patterns for collaborative problem solving [10, 21, 32]. These systems improve performance mainly through external orchestration, such as role decomposition, protocol design, and tool integration. Our work is complementary: rather than modifying only the interaction protocol, we focus on internal partner modeling under partial observability by learning a latent belief state and using it to shape downstream control.

Interactive environments and agent training. Another line of work develops environments and training pipelines for interactive agents. SOTOPIA [36] provides open-ended, multi-turn interactions with private goals and multidimensional evaluation, and subsequent work studies imitation and self-reinforcement, utterance-level reward design, and scalable simulation in this setting [36, 30, 34, 35]. In the multimodal setting, MMROLE introduces a framework for developing and evaluating multimodal role-playing agents using persona-grounded images, dialogues, and reward-model-based evaluation [4]. These works provide important testbeds and training recipes for interactive agents, but they generally do not learn an explicit amortized partner model that is reused as a decision variable across benchmarks. Our approach instead learns a latent partner representation from interaction history, couples it with reward modeling, and distills the resulting signal into a policy.

Theory of Mind and agent modeling. ToM aims to infer or represent agents' latent mental states (beliefs, goals, intentions) to predict future behavior and mental states. ToMnet [22] learns latent models of agents from behavioral traces to predict future actions and agent-specific characteristics. In language-only settings, ToMi, BigToM, ToMBench, and OpenToM provide increasingly controlled or comprehensive evaluations of belief tracking and mental-state reasoning [15, 7, 2, 33]. SymbolicToM shows that explicit belief tracking can substantially improve ToM reasoning at decoding time [24]. In multimodal settings, MMToM-QA and MuMA-ToM extend ToM evaluation beyond text, and MuMA-ToM further introduces LIMP, a model-based inverse multi-agent planning approach for multimodal mental-state inference [11, 27]. Relative to this literature, our contribution is not a new

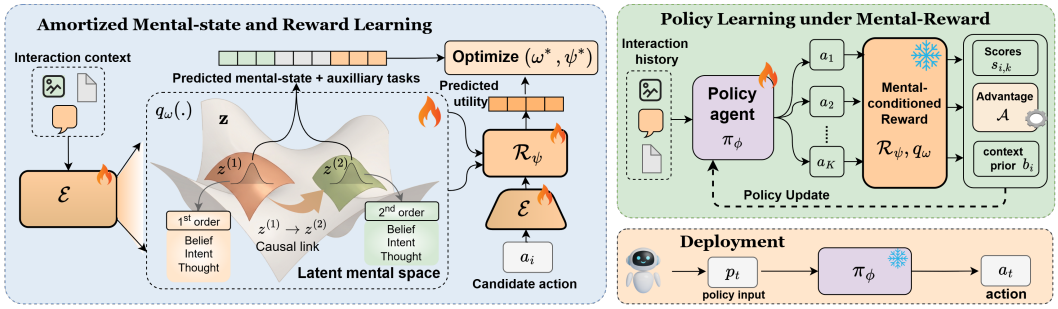


Figure 2: **Method overview.** Phase 1 jointly learns an amortized recursive mental-model and a mental-conditioned reward model from multi-agent context. The latent contains first- and second-order belief, intent, and thought subspaces, which condition reward prediction. Phase 2 freezes the mental-state encoder and reward model, scores sampled policy actions with the learned mental reward, and optimizes the policy with mental-reward guidance. Phase 3 deploys only the trained policy in the multi-agent setting.

ToM benchmark or a task-specific inference procedure. Instead, we learn an amortized latent mental state from interaction history, couple it to utility prediction, and use the resulting signal to train policies for general multi-agent interaction across both language-only and multimodal benchmarks.

3 Methodology

We study multi-agent interaction under partial observability. As shown in Fig. 2, we first learn an amortized latent mental-model of a counterpart's hidden state together with a utility model over candidate actions. We then use this learned signal to train a standalone policy. Finally, we deploy the policy directly at test time. We present the detailed mathematical formulation below. For notational clarity, we present the two-agent case with one acting agent and one counterpart.

3.1 Amortized Mental-model Learning

We consider sequential interaction over a finite horizon T. At time t, the acting agent receives an observation o_t, maintains the interaction history $h_t = (o_1, a_1, \ldots, o_t)$, and selects an action $a_t \in \mathcal{A}$. The action may be free-form language or a structured environment action. In multimodal settings, o_t may also contain visual or other non-textual signals.

The effect of a_t depends not only on the external environment, but also on latent mental properties of the counterpart, such as goals, beliefs, knowledge, capabilities, and strategies. We denote these hidden variables by z_t. A Bayes-optimal agent would maintain the posterior $p(z_t \mid h_t)$, but exact inference is generally intractable in open-ended interaction. We therefore learn an amortized mental-state model

$$q_\omega(z_t \mid h_t) \approx p(z_t \mid h_t), \tag{1}$$

which maps the observable history to a compact latent belief state.

However, ToM is inherently multi-order (as illustrated in Figure 4d). It requires reasoning not only about a partner's first-order state, but also about that partner's model of agent or environment. We capture this recursion by factorizing the mental-model latent as

$$q_\omega(z_t^{(1)}, z_t^{(2)} \mid h_t) = q_{\omega_1}(z_t^{(1)} \mid h_t)\, q_{\omega_2}(z_t^{(2)} \mid h_t, z_t^{(1)}). \tag{2}$$

Here $z_t^{(1)}$ represents the first-order partner state, while $z_t^{(2)}$ represents higher-order state conditioned on the first-order inference. Each mental-latent $z_t^{(\ell)}$ is decomposed into semantically meaningful subspaces representing belief, intent and thought-process, which are the fundamental elements of human theory-of-mind

$$z_t^{(\ell)} = \left[z_{t,\text{belief}}^{(\ell)}; z_{t,\text{intent}}^{(\ell)}; z_{t,\text{thought}}^{(\ell)}\right], \qquad \ell \in \{1, 2\} \tag{3}$$

Using a single flat latent to represent both first- and second-order states can conflate two distinct estimation problems (Figure 4c,d). The recursive factorization keeps them connected without

collapsing them. Practically, each latent factor is modeled as a multi-variate Gaussian distribution,

$$q_{\omega_1}(z_t^{(1)} \mid h_t) = \mathcal{N}\big(\mu_1(\mathcal{E}(h_t)), \mathrm{diag}(\sigma_1^2(\mathcal{E}(h_t)))\big) \tag{4}$$

$$q_{\omega_2}(z_t^{(2)} \mid h_t, z_t^{(1)}) = \mathcal{N}\big(\mu_2([\mathcal{E}(h_t); z_t^{(1)}]), \mathrm{diag}(\sigma_2^2([\mathcal{E}(h_t); z_t^{(1)}]))\big) \tag{5}$$

Modeling latent factors as a Gaussian posterior family with a VAE-style parameterization gives the mental model a smooth geometry over beliefs, intents, and thoughts while representing uncertainty through the posterior variance. This lets the model represent a continuous distribution over plausible mental states rather than committing to a single point estimate, which is important because another agent's beliefs, intentions, and strategies are only partially observed and must be inferred under uncertainty. The first-order latent uses a standard normal prior, while the higher-order latent is regularized against a conditional Gaussian prior induced by $z_t^{(1)}$ so that the variational bottleneck preserves the recursive factorization. For source-specific mental-state supervision or proxy evidence y_t, the latent is learned with a compressed inference objective

$$\omega^\star \in \arg\min_{\omega} \mathbb{E}_{(h_t, y_t)\sim\mathcal{D}} \left[\mathcal{D}_{\mathrm{tom}}(q_\omega(z_t \mid h_t),\, y_t) + \beta\, \Omega_{\mathrm{comp}}(q_\omega(z_t \mid h_t))\right], \tag{6}$$

where the discrepancy term $\mathcal{D}_{\mathrm{tom}}$ grounds the latent in available mental-state evidence. The compression penalty Ω_{comp} limits the capacity of the inferred latent state. In our implementation, it is instantiated as a recursive variational bottleneck with a Gaussian posterior, a history-independent prior for $z_t^{(1)}$, and a conditional prior for $z_t^{(2)}$.

3.2 Mental-Model-Aware Utility

A latent mental-model on its own cannot act as a decision variable and therefore cannot inform the agent what to do. To translate mental-model into action, we need a way to evaluate candidate actions against the inferred counterpart state. We therefore learn a reward utility function that scores arbitrary candidates conditioned on the mental-model latent.

For an evaluation setting e, let $U_e(h_t, a_t, \theta_t) \in \mathbb{R}^{D_e}$ be the outcome vector for action a_t under the hidden counterpart state, and let $w_e \in \Delta^{D_e-1}$ be the scalarization of those dimensions. For example, an open-ended social setting may score goal completion and social quality, while a single-step ToM setting may score answer correctness. We learn a mental-model-conditioned utility predictor

$$\bar{\mathbf{u}}_{\omega,\psi}(h_t, a_t) = \mathbb{E}_{z_t\sim q_\omega(\cdot\mid h_t)}\left[\hat{\mathbf{u}}_\psi(h_t, a_t, z_t)\right]. \tag{7}$$

where $\hat{\mathbf{u}}_\psi(h_t, a_t, z_t)$ is the outcome from the utility head $\mathcal{R}_\psi$. This expectation turns the latent mental model into a score over candidate actions. It asks how good action a_t is on average across the mental states that remain plausible given the history. The supervision used to fit $\mathcal{R}_\psi$ varies across benchmarks. Refer to Appendix A.1 for a detailed explanation.

Learning the mental-model and the utility separately tends to produce two complementary failure modes. A mental-model trained only on mental-state annotations becomes interpretable but decision-irrelevant. It captures what is describable about the partner rather than what is actionable. A reward model trained without pressure to use that latent explains its scores through surface features of the candidate response and ignores partner state entirely. We therefore optimize the two jointly, so that gradients from the reward objective shape the mental-model toward decision-relevance, and the mental-model in turn forces the reward model to route through it.

$$(\omega^\star, \psi^\star) \in \arg\min_{\omega,\psi} \mathbb{E}_{x\sim\mathcal{D}} \left[\underbrace{\mathcal{J}_{\mathrm{rew}}(x;\omega,\psi)}_{\text{candidate reward}} + \underbrace{\mathcal{J}_{\mathrm{mm}}(x;\omega)}_{\text{mental-state inference}} + \underbrace{\mathcal{J}_{\mathrm{aux}}(x;\omega,\psi)}_{\text{auxiliaries}}\right]. \tag{8}$$

$\mathcal{J}_{\mathrm{rew}}$ fits the candidate-action scorer in Eq. (7), including hard-negative ranking when contrastive responses are available. $\mathcal{J}_{\mathrm{mm}}$ anchors z_t to first- and second-order beliefs, intents, thoughts, or other mental-state evidence through reconstruction. $\mathcal{J}_{\mathrm{aux}}$ collects rationale-conditioned utility prediction and lightweight source-specific regularizers such as future-action prediction. The compact objective and dataset-specific instantiations are given in Appendix A.1.

3.3 Policy Learning under Mental-conditioned Reward

To integrate the learned mental-model into a deployable agent, we use it as a training signal through the utility. Specifically, the policy must learn to map observable history directly to actions that the

mental-model-aware utility would prefer. Through this training, the policy parameters internalize which observable contexts call for which mental-state-sensitive actions.

For the evaluation scalarizer $w_e \in \Delta^{D_e-1}$, the mental-model-aware utility induces a scalar reward

$$r_{\omega^\star,\psi^\star}(h_t, a_t) = w_e^\top \bar{\mathbf{u}}_{\omega^\star,\psi^\star}(h_t, a_t). \tag{9}$$

Policy learning then proceeds through candidate generation and mental-model-aware scoring. For a history h_t, the current policy proposes an action conditioned on $p_t = \phi_{\mathrm{pol}}(h_t)$; the scorer evaluates that action using Eq. (9); and the policy update increases the likelihood of actions whose reward is high relative to other plausible choices in the same context. In implementation, we instantiate this update with mental-reward-guided GRPO, using $r_{\omega^\star,\psi^\star}$ to score sampled candidates and form group-relative advantages (Fig. 2). A compact objective for this update is

$$\phi^\star \in \arg\max_{\phi} \mathbb{E}_{\tau\sim\pi_\phi}\left[\sum_t r_{\omega^\star,\psi^\star}(h_t, a_t) - \eta_{\mathrm{ref}}\,\mathrm{KL}(\pi_\phi(\cdot \mid p_t) \,\|\, \pi_{\mathrm{ref}}(\cdot \mid p_t))\right]. \tag{10}$$

The reference term keeps the update near the base policy while the mental-model-aware reward changes the relative preference among plausible actions. In open-ended episodes, the candidates are sampled responses. In structured ToM benchmarks, the same signal scores answer options or constrained outputs.

The key mechanism is gradient assignment. When an action receives high $r_{\omega^\star,\psi^\star}$, the update increases $\log \pi_\phi(a_t \mid p_t)$ in contexts sharing the observable cues in h_t. Because that reward was computed by marginalizing over $q_{\omega^\star}(z_t \mid h_t)$, the gradient carries information about the hidden mental state. Over training, the policy absorbs regularities linking observable cues to partner beliefs, intents, and goals, and the actions the critic prefers under them.

After training, the deployment uses only the learned policy

$$a_t \sim \pi_{\phi^\star}(\cdot \mid p_t). \tag{11}$$

Detailed architectural design, component choices, dataset-specific instantiations and implementation hyperparameters are deferred to Appendix A.1.

4 Experimentation

Implementation details. All trained open-source agents use LoRA adapters on 7B-scale language or vision-language backbones. We use Qwen2.5-7B-Instruct as the primary language backbone, including LLaMA2-7B-Chat and Mistral-7B-Instruct-v0.3 as additional SOTOPIA comparisons, and use Qwen2.5-VL-Chat for multimodal experiments on MMRole. For SOTOPIA, we augment SOTOPIA-π[30] episodes with ToM labels and utility annotations to train the mental-state/reward model; for BigToM and ToMi, we train on 5K synthetic BigToM-style samples and evaluate on the official BIGTOM and TOMI splits; for MMROLE, we augment the released training set with mental labels. CRAIGSLIST-BARGAIN and FANTOM are used only as transfer evaluations. We first train the mental-state/reward model and then optimize the policy with supervised warmup when used, followed by mental-reward-guided GRPO using rewards from the frozen scorer. Detailed data construction, sample counts, latent dimensions, optimizer settings, decoding parameters, and hardware are reported in Appendix A.1.3; Appendix Table 10 summarizes the augmented supervision, and Appendix A.2.4 reports automated filtering and blinded human verification.

Baselines and evaluation. For SOTOPIA, we compare against zero-shot/base LLaMA2-7B, Mistral-7B, and Qwen2.5-7B agents, SFT and GRPO variants trained on our generated data, and closed-source GPT-4o-mini/Gemini baselines, evaluating with the official episode-level seven-dimension SOTOPIA scoring protocol on all, hard, and partner-averaged settings. Craigslist-Bargain uses the released negotiation protocol and reports deal, walkaway, and no-agreement rates. For BigToM and ToMi, we follow the official evaluation scripts and report the released TB/FB/TB$\wedge$FB, Average, Joint, and false-belief split metrics against GPT-style and classic ToMi baselines where available. For MMRole, we compare with the official closed-source/open-source VLM and MMRole-Agent baselines and report the benchmark's normalized role-playing dimensions and overall score.

Table 1: Aggregated performance on SOTOPIA-ALL episodes with a fixed GPT-4o-mini partner.

Dimension	Range	LLaMA		Mistral		Qwen	
		Base	Ours	Base	Ours	Base	Ours
Social rules	$[-10, 0]$	-0.200	-0.167	-0.122	-0.189	-0.144	-0.122
Secret	$[-10, 0]$	-0.511	-0.256	-0.156	-0.111	-0.144	-0.133
Financial	$[-5, 5]$	0.522	0.622	0.489	0.833	0.333	0.844
Relationship	$[-5, 5]$	2.656	2.844	3.022	3.178	2.656	3.533
Knowledge	$[0, 10]$	5.333	5.933	5.722	6.456	5.144	6.611
Believability	$[0, 10]$	8.678	8.722	8.787	8.856	8.722	8.933
Goal	$[0, 10]$	6.833	7.089	6.833	7.467	6.467	7.322
Overall		3.330	**3.541**	3.511	**3.784**	3.290	**3.856**

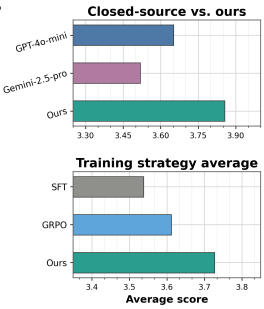


Figure 3: SOTOPIA average-score comparisons. **Top:** closed-source models versus ours. **Bottom:** SFT, GRPO, and ours averaged over open-source models.

Table 2: Selected comparison on SOTOPIA-HARD episodes with GPT-4o-mini partner.

Model	Method	Goal	Bel.	Know.	Avg.
LLaMA	Base	5.000	8.571	4.071	5.880
	Ours	5.571	8.500	4.429	**6.166**
Mistral	Base	5.071	8.429	6.643	6.714
	Ours	5.714	8.714	5.786	**6.738**
Qwen	Base	5.643	8.643	4.857	6.381
	Ours	5.643	9.000	5.714	**6.785**

Table 3: Partner-averaged (GPT-4o-mini and self) scores on selected SOTOPIA dimensions.

Model	Method	Goal	Bel.	Know.	Avg.
LLaMA	Base	6.333	8.561	4.872	6.588
	Ours	7.045	8.739	5.950	**7.245**
Mistral	Base	6.878	8.766	5.683	7.109
	Ours	7.273	8.850	6.534	**7.552**
Qwen	Base	6.773	8.811	5.678	7.087
	Ours	7.317	8.945	6.639	**7.633**

Table 4: Craigslist-bargain generalization transfer statistics.

Model	Method	Deal	Walk.	No Agr.
Closed-Source				
GPT-4o-mini		96.0%	3.0%	1.0%
Open-Source (7B)				
LLaMA2	Base	93.0%	7.0%	0.0%
	Ours	94.0%	1.0%	5.0%
Qwen2.5	Base	91.0%	7.0%	2.0%
	Ours	**94.0%**	**5.0%**	**1.0%**

4.1 Main Experimentation Results

Results on SOTOPIA. Table 1 reports the main SOTOPIA-ALL evaluation results under a fixed GPT-4o-mini partner. Our method improves the overall score for all three open-source agents (LLaMA: 3.330→3.541, Mistral: 3.511→3.784, Qwen: 3.290→3.856), indicating that the gains are not specific to one model family. The pronounced improvements on task-facing dimensions such as Knowledge and Goal, with high and stable Believability, suggests that the learned mental-model signal improves goal-directed reasoning without making the agents less plausible. Table 2 shows the same pattern on SOTOPIA-HARD, where average scores improve for all three LLM agents, showing that the method continues to help when scenarios demand more difficult partner modeling.

Table 3 evaluates generalization across partner types by averaging results over the GPT-4o-mini and self-partner settings. The improvement is consistent across all agents (LLaMA: $6.588 \rightarrow 7.245$, Mistral: $7.109 \rightarrow 7.552$, and Qwen: $7.087 \rightarrow 7.633$). These gains suggest that the method does not depend on a particular partner model and transfers across both closed-source and open-source counterparts.

Figure 3 summarizes the aggregate SOTOPIA trends. Our 7B Qwen2.5 agent with explicit mental-model training outperforms stronger closed-source baselines in the top panel, while the bottom panel shows consistent gains over SFT-only and GRPO-only variants (trained on our ToM-augmented data) averaged across LLaMA, Mistral, and Qwen. This suggests that explicit partner-state modeling contributes beyond standard supervised or reward-only training.

Finally, Table 4 evaluates cross-domain transfer. We evaluate the agents trained exclusively with SOTOPIA-derived ToM data on a random subset of 100 CRAIGSLIST BARGAIN negotiation examples without any task-specific fine-tuning. The models retain high deal rates and reduce walkaways. Together, Tables 1–4 and Figure 3 suggest that explicit partner-state modeling improves both in-domain social interaction and out-of-domain negotiation transfer.

Results on BIGTOM & TOMI. Tables 5 & 7 evaluate transfer from open-ended dialogue to controlled ToM reasoning. BIGTOM (Table 5) is especially diagnostic because TB∧FB only rewards a model when both branches of the same scenario are correct. The model must maintain two counterfactual mental states for the same agent. Base models often handle the easier true-belief branch but lose consistency once the false-belief branch or initial belief changes. SFT on our enriched data improves, but it still treats the output largely as a supervised answer pattern. In contrast, our mental-model variant is more consistent across forward belief, forward action, and backward belief because of its ability to encode higher order hidden states (further comparisons Appendix Table 13). Table 7 provides an OOD generalization check on TOMI, whose structure and metrics differ from the BIGTOM training format. We cross-evaluate our policy trained with BIGTOM-style data on TOMI and observe that it improves Average and Joint scores over the base model. The classic ToMi

Table 5: **BIGTOM results.** TB/FB/TB∧FB denote true-belief, false-belief, and paired-branch accuracy without/with the initial belief sentence. Bold marks the best open-source result.

Task	Model	Without Initial Belief			With Initial Belief		
		TB	FB	TB∧FB	TB	FB	TB∧FB
Forward Belief	GPT-4	99.0	98.0	97.0	91.0	99.0	90.0
	GPT-4 (CoT)	99.0	99.0	98.0	99.0	99.0	98.0
	Base	83.0	94.0	77.0	49.5	98.5	48.0
	SFT	95.5	88.5	84.5	94.0	96.5	92.0
	Ours	**97.0**	**98.0**	**96.0**	**97.5**	**99.5**	**97.5**
Forward Action	GPT-4	98.0	81.0	79.0	98.0	92.0	90.0
	GPT-4 (CoT)	99.0	88.0	87.0	99.0	96.0	95.0
	Base	99.5	64.0	63.5	**99.5**	68.0	67.5
	SFT	93.0	82.5	76.5	92.0	80.0	73.0
	Ours	**99.5**	**79.0**	**78.5**	99.0	**81.0**	**80.0**
Backward Belief	GPT-4	86.0	53.0	40.0	62.0	77.0	40.0
	GPT-4 (CoT)	84.0	54.0	38.0	76.0	63.0	40.0
	Base	85.0	92.5	77.5	53.5	99.0	52.5
	SFT	94.5	90.5	85.0	94.5	95.5	91.0
	Ours	**97.0**	**98.5**	**95.5**	**98.5**	**99.5**	**98.5**

Table 6: **MMROLE results.** Official normalized role-play metrics are averaged into Overall. Blue rows mark ours, which improves over MMRole-Agent overall and on In/Out splits.

MRPAs	IA	Flu	Coh	ITR	RA	PC	KC	TC	Overall
GPT-4 Turbo	1.055	1.032	1.084	1.097	1.092	1.168	1.103	1.161	1.099
Gemini Pro Vision	0.999	1.007	1.028	1.009	1.013	1.052	1.013	1.050	1.021
Claude 3 Opus	1.127	1.070	1.149	1.167	1.146	1.219	1.168	1.213	**1.157**
QWen-VL-Max	1.014	1.012	1.035	1.034	1.029	1.042	1.021	1.041	1.028
QWen-VL-Chat	0.844	0.954	0.879	0.850	0.829	0.778	0.827	0.785	0.843
LLaVA-NeXT-Mistral-7B	0.948	0.986	0.964	0.938	0.933	0.924	0.940	0.921	0.944
Yi-VL-6B	0.844	0.919	0.859	0.828	0.811	0.776	0.820	0.774	0.829
MMRole-Agent	0.998	1.000	0.997	0.993	0.987	1.000	0.992	0.988	0.994
Ours	1.013	1.001	1.043	1.010	1.031	1.032	1.031	1.021	**1.023**
MMRole-Agent (In)	1.000	1.000	0.999	0.997	0.989	1.012	0.997	0.997	0.999
MMRole-Agent (Out)	0.992	0.999	0.993	0.979	0.981	0.963	0.977	0.962	0.981
Ours (In)	1.014	1.000	1.044	1.008	1.031	1.040	1.033	1.024	**1.024**
Ours (Out)	1.009	1.004	1.041	1.017	1.030	1.008	1.026	1.015	**1.019**

Model	Average	Joint	w/o FB 1st	w/o FB 2nd	w/o FB Reality	w/o FB Memory	FB 1st	FB 2nd
Classic TOMi baselines								
MemNN	77.20	44.30	85.45	82.67	93.39	98.90	12.62	17.27
RelNet	86.00	57.40	96.42	95.37	100.0	99.90	10.40	17.81
EntNet	**90.60**	66.80	94.29	85.08	100.0	100.0	54.95	36.55
Closed source LLMs								
GPT-3.5-Instruct	79.78	36.94	94.42	84.05	97.30	94.79	34.26	7.82
GPT-4o-mini	71.00	24.42	77.92	12.57	99.40	100.0	81.68	92.73
Base	87.92	55.16	91.09	78.45	99.40	99.80	74.26	71.27
Ours	89.16	60.46	93.16	91.09	94.79	98.20	66.34	62.55

Table 7: **TOMI Generalization** from the BIGTOM-trained policy. Ours improves the aggregate Average and Joint scores over the base model and strengthens non-false-belief first-/second-order belief tracking, highlighting OOD generalization of our mental-model trained policy.

baselines remain strong because they are task-specific models fit to ToMi's synthetic state-tracking format and do not have broader multi-agent generality. Further, our model's high score on w/o FB second-order questions shows that the transferred latent is capturing nested belief structure when no false-belief conflict is introduced. Overall, the gains suggest that the model learns a reusable hidden-state interface for ToM.

Results on MMROLE. Table 6 shows that our multimodal mental-model training improves over the released MMROLE-Agent baseline on the overall score, both in-domain and out-of-domain. The gain is most visible on the Out split, where ours reaches 1.019 against 0.981 for MMROLE-Agent (Out), indicating that the learned mental state does not merely fit seen character-image pairs but also transfers to held-out role contexts. The strongest improvements appear on Coherence, Response Accuracy, Personality Consistency, Knowledge Consistency, and Tone Consistency, which are precisely the dimensions where role-play depends on more than image recognition or fluent captioning. These metrics require inferring who the character is, what they know, how they should speak, and how the visual scene relates to the current dialogue.

5 Analysis and Ablations

Is mental-model simply a compression code? We train a flat compression VAE on the same ToM-augmented SOTOPIA-π contexts and responses, and compare it our recursive mental-model. We fit lightweight probes from z to the reward head. We then scramble z across held-out examples while keeping the reward head fixed. Figure 4a shows that scrambling the flat compression VAE latent has only a weak effect on reward-vector MSE, while scrambling mental-model z sharply disrupts joint reward prediction. Thus, the flat latent can store reconstructive mental-summary information, but the mental-model z latent is more tightly coupled to reward-relevant decision structure.

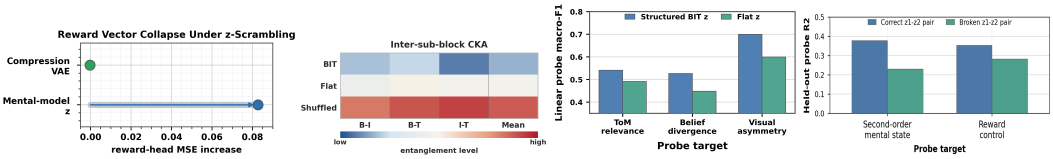
Significance of BIT structure. Figure 4b probes the internal structure of the belief, intent, and thought (BIT) slices using pairwise CKA on SOTOPIA data. The BIT variant produces subspaces that are separated yet related, whereas the model trained with flat mental-summary including shuffled variant shows significant inter-slice similarity. Figure 4c further shows that linear probes recover ToM-relevant MMROLE factors more accurately from BIT than from a flat latent of the same combined

Table 8: Sensitivity to mental-state supervision on SOTOPIA with Qwen2.5-7B. Higher is better.

Mental supervision	Overall score
0% (baseline)	3.300
25%	3.650
50%	3.750
100%	**3.856**

Table 9: Posterior-family ablation on a held-out, scenario-level BIGTOM split (n=5,976). Accuracy and probe scores are percentages.

Mental-model posterior	Reward acc.	Margin	Belief probe
Diagonal Gaussian (ours)	96.6	1.05	99.8
Full covariance, low-rank (diag + $UU^\top$)	96.8	1.05	99.9



a) b) c) d)

Figure 4: a) z-scrambling barely affects the compression VAE but sharply increases BIT reward-vector MSE, showing stronger reward-causal structure. b) Pairwise CKA shows complementary belief, intent, and thought slices. c) Frozen-latent probes recover ToM-relevant MMROLE factors more separably from BIT than from a flat latent. d) Shuffling z_2 weakens second-order probes, supporting the recursive $z_1 \to z_2$ structure.

dimensionality. Together, these diagnostics support the intended interpretation that BIT slices are complementary components rather than redundant copies.

Recursive $z_1 \to z_2$ structure. Figure 4d tests whether the relation between first- and second-order latents carries intrinsic mental-state information, rather than simply adding another independent code. We compare probes on the true example-wise pair (z_1, z_2) against samples where z_2 is shuffled while preserving its marginal distribution. This keeps the amount of latent capacity and the statistics of each block largely unchanged, but breaks the recursive alignment between a partner state and the higher-order state conditioned on it. The shuffled-pair control loses substantially more recoverable second-order mental-state information than reward information, suggesting that the $z_1 \to z_2$ link specifically carries nested ToM structure rather than only generic reward-relevant compression.

Sensitivity to mental-state supervision. We study whether the gains also depend on the available mental-state supervision. We train Qwen2.5-7B with nested 25% and 50% subsets of the SOTOPIA annotations (the 25% subset is contained in the 50% subset) while keeping the mental-model architecture fixed; 100% uses all annotations, whereas 0% is the mental-free reward baseline because no first- or second-order mental targets are supplied. Every condition uses the same evaluation seed, GPT-4o-mini partner, GPT-4o judge, and official SOTOPIA scoring protocol. Table 8 shows that only 25% supervision already improves the overall score by 0.350 points over the baseline, and performance increases with annotation coverage to 3.856. Thus, the method indeed benefits from limited supervision rather than requiring exhaustive annotation.

Posterior-family ablation. We study whether our proposed diagonal posterior loses important within-level correlations among belief, intent, and thought coordinates. We retrain the Stage-1 mental-state/reward model on BIGTOM, holding all settings fixed and replacing only the diagonal Gaussian with a structured full-covariance posterior, $\Sigma = \mathrm{diag}(\sigma^2) + UU^\top$. On the held-out scenario-level split (n=5,976), pairwise reward accuracy and margin measure preferred-versus-negative ranking, while the belief probe measures mental-state recoverability from the latent. Table 9 shows nearly identical reward ranking, margin, and belief recovery; the 0.2-point accuracy difference is within the $\pm$0.65-point sampling uncertainty at this n. The more costly covariance therefore provides no measurable benefit. Cross-coordinate dependence remains available through the shared nonlinear history encoder, the conditional $q(z_2 \mid h, z_1)$, and the joint prediction heads; the diagonal assumption restricts only residual posterior noise.

Functional Latent Traversals. We test whether recovered latent directions correspond to meaningful mental and strategic variables, rather than only post-hoc probe artifacts. For the belief analysis, we linearly probe a held-out branch label indicating whether an agent is aware or not-aware of an event, then perturb near-boundary anchors along the resulting direction $\hat{u}_{\text{belief}}$ and compare against

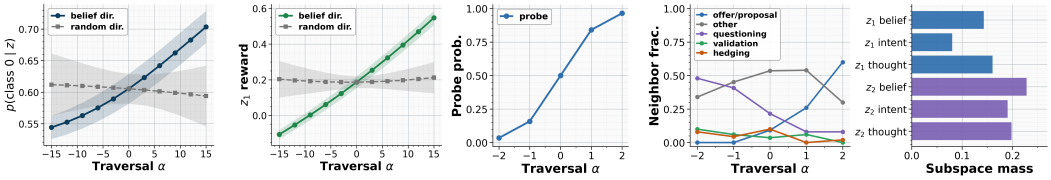


Figure 5: Latent traversal analyses. a) Belief-axis traversal shifts both the branch classifier and frozen reward head beyond random controls. b) Offer/proposal traversal increases strategy signal and local label enrichment, with stronger second-order contribution.

magnitude-matched random directions. Figure 5a shows that this traversal monotonically shifts both the branch classifier and z_1-only reward head, even though the reward head was not trained on the branch label. For the strategy analysis (Figure 5b), we fit a linear probe for the annotated SOTOPIA strategy label *offer/proposal* and move near-boundary latent points along this direction with traversal coefficient α. This is not an edit to one generated response; it tests how the local latent neighborhood changes as α moves toward the strategy concept. In Figure 5b, left panel shows the offer/proposal probe score increasing with α, the middle panel shows nearest-neighbor labels shifting from questioning/other toward offer/proposal, and the right panel decomposes the direction into squared-weight mass across BIT blocks. The larger mass in second-order ToM blocks suggests that proposal-like strategies rely more on modeling how the partner may respond.

6 Conclusion

We introduced a mental-model-enabled agent framework for multi-agent systems, where an agent learns an amortized recursive Theory-of-Mind latent state and couples it to a belief-conditioned reward model. The resulting reward signal is used to train policies that act independently at test time while retaining the benefits of explicit partner-state modeling. Across SOTOPIA, negotiation transfer, BIGTOM/TOMI/FANTOM, and MMROLE, the same framework improves open-source agents on social-interaction and Theory-of-Mind benchmarks. Our results also show that augmenting existing datasets with mental-state annotations, utility signals, rationales, and contrastive negatives provides useful supervision for learning reusable partner representations.

Limitations and future work. Our training environments are primarily dyadic; the zero-shot FANTOM transfer in Appendix A.3 provides initial multi-party evidence, while end-to-end multi-party training with participant-specific latent updates remains future work. The same formulation can also be extended beyond language and image-grounded role play to embodied and robotic settings, where partner beliefs, goals, visibility, and physical constraints evolve over longer horizons. Finally, scaling to larger policies, more diverse partner models, richer online belief updates, and human-in-the-loop evaluation would help test how far explicit mental-state structure continues to improve robust multi-agent behavior.

7 Acknowledgments and Disclosure of Funding

The authors received no external funding, third-party financial support, or specific grants for this work. This research was supported entirely by internal institutional / advisor resources from the authors' affiliated institution. The authors declare no competing financial or non-financial interests.

References

[1] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph,

Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[2] Zhuang Chen, Jincenzi Wu, Jinfeng Zhou, Bosi Wen, Guanqun Bi, Gongyao Jiang, Yaru Cao, Mengting Hu, Yunghwei Lai, Zexuan Xiong, and Minlie Huang. ToMBench: Benchmarking theory of mind in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15959–15983, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[3] Kenneth J. W. Craik. *The Nature of Explanation*. Cambridge University Press, 1943.

[4] Yanqi Dai, Huanran Hu, Lei Wang, Shengjie Jin, Xu Chen, and Zhiwu Lu. MMRole: A comprehensive framework for developing and evaluating multimodal role-playing agents. In *The Thirteenth International Conference on Learning Representations*, 2025.

[5] Prashant Doshi and Piotr J. Gmytrasiewicz. A framework for sequential planning in multi-agent settings, 2011.

[6] Prashant Doshi, Yifeng Zeng, and Qiongyu Chen. Graphical models for interactive POMDPs: representations and solutions. *Autonomous Agents and Multi-Agent Systems*, 18(3):376–416, 2009.

[7] Kanishk Gandhi, Jan-Philipp Fraenken, Tobias Gerstenberg, and Noah Goodman. Understanding social reasoning in language models with language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 13518–13529. Curran Associates, Inc., 2023.

[8] Alison Gopnik and Henry M. Wellman. Why the child's theory of mind really is a theory. *Mind & Language*, 7(1-2):145–171, 1992.

[9] He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. Decoupling strategy and generation in negotiation dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2333–2343, 2018.

[10] Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*, 2023.

[11] Chuanyang Jin, Yutong Wu, Jing Cao, Jiannan Xiang, Yen-Ling Kuo, Zhiting Hu, Tomer Ullman, Antonio Torralba, Joshua B. Tenenbaum, and Tianmin Shu. MMToM-QA: Multimodal theory of mind question answering. *arXiv preprint arXiv:2401.08743*, 2024.

[12] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998.

[13] John F. Kihlstrom and Nancy Cantor. Social intelligence. In *The Cambridge Handbook of Intelligence*, pages 756–779. Cambridge University Press, 2 edition, 2020.

[14] Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Le 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*, 2023.

[15] Matthew Le, Y-Lan Boureau, and Maximilian Nickel. Revisiting the evaluation of theory of mind through question answering. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5872–5877, Hong Kong, China, November 2019. Association for Computational Linguistics.

[16] Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: Communicative agents for "mind" exploration of large language model society. *arXiv preprint arXiv:2303.17760*, 2023.

[17] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. Dailydialog: A manually labelled multi-turn dialogue dataset, 2017.

[18] Ansong Ni, Ruta Desai, Yang Li, Xinjie Lei, Dong Wang, Jiemin Zhang, Jane Yu, Ramya Raghavendra, Gargi Ghosh, Shang-Wen Li, and Asli Celikyilmaz. Collaborative reasoner: Self-improving social agents with synthetic conversations. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[19] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 2022.

[20] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22. Association for Computing Machinery, 2023.

[21] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. ChatDev: Communicative agents for software development. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15174–15186, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[22] Neil C. Rabinowitz, Frank Perbet, H. Francis Song, Chiyuan Zhang, S. M. Ali Eslami, and Matthew Botvinick. Machine theory of mind. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4218–4227. PMLR, 10–15 Jul 2018.

[23] Maarten Sap, Ronan Le Bras, Daniel Fried, and Yejin Choi. Neural theory-of-mind? on the limits of social intelligence in large lms. arXiv preprint, 2022. Originally published at EMNLP 2022; arXiv version revised 2023.

[24] Melanie Sclar, Sachin Kumar, Peter West, Alane Suhr, Yejin Choi, and Yulia Tsvetkov. Minding language models' (lack of) theory of mind: A plug-and-play multi-character belief tracker. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13960–13980, Toronto, Canada, July 2023. Association for Computational Linguistics.

[25] Zhihong Shao, Peiyuan Wang, Qidong Zhu, Runxin Xu, Junjie Song, Xiaokang Bi, Haowei Zhang, Mingchuan Li, Y. K. Wu, Jie Yu, et al. Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[26] Natalie Shapira, Mosh Levy, Seyed Hossein Alavi, Xuhui Zhou, Yejin Choi, Yoav Goldberg, Maarten Sap, and Vered Shwartz. Clever hans or neural theory of mind? stress testing social reasoning in large language models. arXiv preprint, 2023.

[27] Haojun Shi, Suyu Ye, Xinyu Fang, Chuanyang Jin, Leyla Isik, Yen-Ling Kuo, and Tianmin Shu. MuMA-ToM: Multi-modal multi-agent theory of mind. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(2):1510–1519, 2025.

[28] Michael Tomasello. *Becoming Human: A Theory of Ontogeny*. Belknap Press, 2021.

[29] Tomer Ullman. Large language models fail on trivial alterations to theory-of-mind tasks. arXiv preprint, 2023.

[30] Ruiyi Wang, Haofei Yu, Wenxin Zhang, Zhengyang Qi, Maarten Sap, Graham Neubig, Yonatan Bisk, and Hao Zhu. Sotopia-π: Interactive learning of socially intelligent language agents. arXiv preprint, 2024.

[31] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. Ethical and social risks of harm from language models. arXiv preprint, 2021.

[32] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. In *COLM*, 2024.

[33] Hainiu Xu, Runcong Zhao, Lixing Zhu, Jinhua Du, and Yulan He. OpenToM: A comprehensive benchmark for evaluating theory-of-mind reasoning capabilities of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8593–8623, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[34] 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.

[35] Xuhui Zhou, Zhe Su, Sophie Feng, Jiaxu Zhou, Jen-tse Huang, Hsien-Te Kao, Spencer Lynch, Svitlana Volkova, Tongshuang Wu, Anita Woolley, Hao Zhu, and Maarten Sap. SOTOPIA-s4: a user-friendly system for flexible, customizable, and large-scale social simulation. In Nouha Dziri, Sean (Xiang) Ren, and Shizhe Diao, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 350–360, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.

[36] Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. Sotopia: Interactive evaluation for social intelligence in language agents. In *The Twelfth International Conference on Learning Representations*, 2024. ICLR 2024 spotlight. Also available as arXiv:2310.11667.

Mental-Models for Multi-Agent Systems

A Appendix

The appendix is organized as below:

- **A.1 Methodology, architecture, and training details** (Sec. A.1)
- **A.2 Dataset-specific instantiations** (Sec. A.2)
- **A.3 Additional results and ablations** (Sec. A.3)
- **A.4 Impact Statement** (Sec. A.4)

A.1 Methodology, Architecture, and Training Details

A.1.1 Dataset-Specific Methodology and Notation

The main paper presents the benchmark-agnostic version of our method. Here we give the concrete instantiations used for each dataset. Across all benchmarks, an example is represented as a tuple

$$(c_i, a_i^+, a_i^-, \mathbf{u}_i, \boldsymbol{\gamma}_i, m_i^{(1)}, m_i^{(2)}, \xi_i), \tag{12}$$

where c_i is the observable context, a_i^+ is the observed or preferred response, a_i^- is an optional hard negative, $\mathbf{u}_i$ is a dataset-specific utility vector, $\boldsymbol{\gamma}_i$ is an optional dimension mask indicating which utility dimensions a contrastive negative is intended to violate, $m_i^{(1)}$ and $m_i^{(2)}$ are first- and second-order mental-state targets when available, and ξ_i is optional rationale or reward-explanation text. Missing fields are handled by masking the corresponding auxiliary losses, which lets the same objective support open-ended dialogue, structured belief QA, and multimodal role-play.

A.1.2 Architecture Overview

Stage 1 uses a single causal language-model or vision-language backbone with low-rank adapters and task-specific heads. The backbone encodes the task context c_i and, when scoring a candidate action, the isolated action a_i in separate forward passes, producing pooled context and action states that feed the latent posterior and reward heads. In our implementation, both recursive latent levels use $d_z = 128$ dimensions and are partitioned into belief, intent, and thought subspaces. These subspaces are not required by the general objective, but they provide a stable interface for attaching first-order and second-order mental-state decoders.

A.1.3 Implementation Details

All open-source models are trained with LoRA adapters on 7B-scale backbones. For language-only experiments, Qwen2.5-7B-Instruct is the main policy and reward backbone; LLaMA2-7B-Chat and Mistral-7B-Instruct-v0.3 are included for additional SOTOPIA comparisons. For multimodal experiments, we use Qwen2.5-VL-Chat. The training sources are: SOTOPIA-π episodes augmented with ToM labels and utility annotations, 5K synthetic BIGTOM-style scenarios with first- and second-order mental labels, and MMROLE inter-role training turns augmented with visual ToM labels. TOMI, official BIGTOM, official SOTOPIA, CRAIGSLIST-BARGAIN, and FANTOM are used only for evaluation or transfer checks. SOTOPIA training episodes do not overlap with the original SOTOPIA evaluation episodes.

Across benchmarks, we optionally use supervised warmup or mental-prefix SFT before GRPO. Unless otherwise noted, the latent mental state uses $d_z = 128$ with belief, intent, and thought subspaces. Training uses bf16 mixed precision, AdamW, warmup scheduling, KL annealing, and delayed second-order latent updates. Typical hyperparameter ranges are LoRA rank 8–32, LoRA $\alpha = 16$–64, dropout 0.05, reward-model learning rate 2×10^{-5}, GRPO learning rate 1×10^{-6}–3×10^{-6}, group size 8, clipping coefficient 0.1–0.2, KL coefficient 0.06–0.1, temperature 0.6–0.8, and top-p 0.9–0.95. Experiments were run on NVIDIA RTX A6000 48GB GPUs.

A.1.4 Shared Mental–Reward Objective

Stage 1 learns a *coupled* latent model of mental state and utility. The objective is not to infer hidden state in isolation, nor to fit a reward predictor independently of social reasoning, but to learn a shared latent representation that is simultaneously constrained by *mental-state supervision* and *action-utility supervision*. The datasets differ in what counts as context, response, utility, and mental-state annotation; the loss template below is shared.

A shared language-model or vision-language backbone encodes c_t into a context representation $e(c_t)$ and any candidate action a into an action representation $e(a)$. Dataset-specific subsections below define the concrete form of c_t, $\mathbf{u}_t$, mental-state targets, and negatives.

Recursive amortized mental-state inference. We parameterize the amortized belief with a recursive two-level latent hierarchy,

$$q_\omega(z_t^{(1)}, z_t^{(2)} \mid c_t) = q_{\omega_1}(z_t^{(1)} \mid c_t)\, q_{\omega_2}(z_t^{(2)} \mid c_t, z_t^{(1)}), \tag{13}$$

$$q_{\omega_1}(z_t^{(1)} \mid c_t) = \mathcal{N}\big(\mu_1(e(c_t)), \mathrm{diag}(\sigma_1^2(e(c_t)))\big), \tag{14}$$

$$q_{\omega_2}(z_t^{(2)} \mid c_t, z_t^{(1)}) = \mathcal{N}\big(\mu_2([e(c_t); z_t^{(1)}]), \mathrm{diag}(\sigma_2^2([e(c_t); z_t^{(1)}]))\big). \tag{15}$$

The first latent $z_t^{(1)}$ captures first-order inferences about the counterpart, while the second latent $z_t^{(2)}$ captures higher-order inferences conditioned on $z_t^{(1)}$. In our implementation, each latent may be further structured into semantically designated subspaces corresponding to factors such as belief, intent, and thought, but the methodology only requires that the posterior be recursively factorized.

Belief-aware utility prediction. Given a candidate action a, the main utility head predicts a vector-valued outcome

$$\hat{\mathbf{u}}_t^{\mathrm{joint}}(a) = g_\psi^{\mathrm{joint}}([z_t^{(1)}; z_t^{(2)}; e(a)]) \in \mathbb{R}^D. \tag{16}$$

This head is action-dependent and captures how the predicted utility changes with the candidate action under the inferred mental state.

Mental-state and rationale supervision. When textual mental-state supervision is available, we decode first- and second-order descriptions directly from the latent states:

$$p_{\eta_1}(m_t^{(1)} \mid z_t^{(1)}), \qquad p_{\eta_2}(m_t^{(2)} \mid z_t^{(2)}). \tag{17}$$

In the implementation, the decoders project latent subspaces into learned prefix tokens and decode the mental-state text through cross-attention. These terms encourage $z_t^{(1)}$ and $z_t^{(2)}$ to encode interpretable social structure rather than serving as arbitrary latent codes.

When rationale or reward-explanation text ξ_t is available, a small explanation-conditioned head combines $[z_t^{(1)}; z_t^{(2)}]$ with its encoded tokens $H(\xi_t)$ and predicts the same utility target. This directly tests whether the textual rationale is consistent with the outcome rather than treating it as free-form generation alone.

Training objective. We display only the core terms supported by the leave-one-out analysis:

$$\mathcal{L}_{\mathrm{MR}} = \mathcal{L}_{\mathrm{vec}} + \lambda_p \mathcal{L}_{\mathrm{pref}} + \lambda_m \mathcal{L}_{\mathrm{mental}} + \lambda_e \mathcal{L}_{\mathrm{expl}} + \mathcal{L}_{\mathrm{KL}}. \tag{18}$$

Here $\mathcal{L}_{\mathrm{vec}}$ is Huber regression from the action-conditioned utility head to the dataset target, and $\mathcal{L}_{\mathrm{mental}}$ reconstructs the two supervised mental states:

$$\mathcal{L}_{\mathrm{vec}} = \mathbb{E}\Big[\ell_{\mathrm{Huber}}\Big(\hat{\mathbf{u}}_t^{\mathrm{joint}}(a_t^+), \mathbf{u}_t\Big)\Big], \tag{19}$$

$$\mathcal{L}_{\mathrm{mental}} = -\mathbb{E}\Big[\log p_{\eta_1}(m_t^{(1)} \mid z_t^{(1)}) + \log p_{\eta_2}(m_t^{(2)} \mid z_t^{(2)})\Big]. \tag{20}$$

$\mathcal{L}_{\mathrm{expl}}$ is the corresponding Huber loss from the explanation-conditioned head to $\mathbf{u}_t$. Each supervised term is masked when its target is unavailable.

The preference term ranks the observed or preferred action above a hard negative only on dimensions that the negative was constructed to violate. With $\gamma_{t,d} \in \{0, 1\}$ denoting that mask,

$$\mathcal{L}_{\text{pref}} = \mathbb{E}\left[\mathbf{1}[a_t^- \text{ exists}] \cdot \frac{1}{\sum_{d=1}^{D} \gamma_{t,d} + \varepsilon} \sum_{d=1}^{D} \gamma_{t,d} \,\text{softplus}\left(\hat{u}_{t,d}^{\text{joint}}(a_t^-) - \hat{u}_{t,d}^{\text{joint}}(a_t^+)\right)\right]. \quad (21)$$

When a dataset provides a scalar preference rather than dimension-specific negative metadata, the mask is all ones. For generated hard negatives, the mask prevents fluent or otherwise unviolated dimensions from being penalized solely because the candidate is negative.

The KL term regularizes the first-order posterior toward a unit Gaussian and the second-order posterior toward a learned prior conditioned on $z_t^{(1)}$:

$$\mathcal{L}_{\text{KL}} = \beta_1 \,\text{KL}\Big(q_{\omega_1}(z_t^{(1)} \mid c_t) \,\|\, \mathcal{N}(0, I)\Big) + \beta_2 \,\text{KL}\Big(q_{\omega_2}(z_t^{(2)} \mid c_t, z_t^{(1)}) \,\|\, p_\chi(z_t^{(2)} \mid z_t^{(1)})\Big). \quad (22)$$

where $p_\chi(z_t^{(2)} \mid z_t^{(1)})$ is a diagonal Gaussian conditional prior. This keeps $z_t^{(2)}$ recursively tied to $z_t^{(1)}$ rather than imposing unconditional independence between the two latent levels.

This compact presentation follows our matched-checkpoint SOTOPIA ablation: removing mental-state reconstruction or rationale conditioning reduces reward-regression correlation by 29% and 26%, respectively. Removing hard-negative training has a smaller effect on regression correlation (7%) but lowers hard-negative preference accuracy from 100% to 81%, confirming that it supplies a distinct discriminative signal.

Optional regularizers. We also optionally add latent-only utility predictions $\hat{\mathbf{u}}_t^{(1)} = g_\psi^{(1)}(z_t^{(1)})$ and $\hat{\mathbf{u}}_t^{(12)} = g_\psi^{(12)}([z_t^{(1)}; z_t^{(2)}])$, first-token future-action prediction, and source-specific auxiliary classifiers or probe regressors. These terms provide only small incremental validation gains relative to the core objective above and primarily serve as anti-bypass regularizers or diagnostics; we therefore mention them only in text rather than expanding additional losses. They are activated only when the corresponding annotation is available.

Overall, this stage produces a latent state that is simultaneously *(i)* amortized from history, *(ii)* semantically grounded in explicit mental-state supervision when available, and *(iii)* directly tied to downstream decision quality through coupled utility prediction. Because missing annotations simply mask the corresponding auxiliary terms, the same training objective can be used across heterogeneous datasets with different supervision granularity.

Stabilization. We use three implementation stabilizers when training the recursive latent model: optional pre-warm training for the first-order latent encoder/decoder on external mental-reasoning data, KL annealing with delayed activation for the second-order latent, and an initial stop-gradient from $z^{(1)}$ into $z^{(2)}$ so that first-order reasoning stabilizes before higher-order coupling is introduced.

A.1.5 Policy Optimization Details

After learning the mental-state and reward model, we freeze it and train a policy $\pi_\phi(a \mid p)$, where $p = \phi_{\text{pol}}(h)$ is the policy input. We optionally initialize the policy with supervised fine-tuning on observed actions,

$$\mathcal{L}_{\text{SFT}} = -\sum_i \log \pi_\phi(a_i^+ \mid p_i). \quad (23)$$

Policy training distinguishes the generation prompt p_i from the reward context c_i: the former is optimized for action generation, while the latter is optimized for latent inference and reward estimation. This split prevents reward estimation from becoming tied to prompt-engineering artifacts.

Scalar reward induced by the frozen model. Because the learned utility is vector-valued, we use a scalarization operator

$$\text{Agg}_w(\mathbf{u}) = w^\top \mathbf{u}, \qquad w \in \Delta^{D-1}, \quad (24)$$

to map $\mathbf{u} \in \mathbb{R}^D$ to a scalar control signal. For a sampled action a, we define the response-dependent score

$$r(h, a) = \text{Agg}_w\big(\hat{\mathbf{u}}^{\text{joint}}(h, a)\big), \quad (25)$$

and the context-only latent score

$$b(h) = \frac{1}{2}\left[\mathrm{Agg}_w\big(\hat{\mathbf{u}}^{(1)}(h)\big) + \mathrm{Agg}_w\big(\hat{\mathbf{u}}^{(12)}(h)\big)\right]. \tag{26}$$

The response-dependent term ranks candidate actions. The context-only score $b(h)$ is useful for diagnostics and optional prompt filtering, but it is not added to candidate-level GRPO advantages because it is constant across the K samples for a fixed prompt and would cancel under group normalization. During policy optimization, the frozen mental-state model is evaluated deterministically using posterior means for $z^{(1)}$ and $z^{(2)}$.

Group-relative policy optimization. For each policy input p_i, we sample K candidate actions

$$a_{i,1}, \ldots, a_{i,K} \sim \pi_\phi(\cdot \mid p_i). \tag{27}$$

Their response-level advantages are normalized within the group:

$$A^{\mathrm{resp}}_{i,k} = \frac{r_{i,k} - \mu_i}{\sigma_i + \varepsilon}, \qquad \mu_i = \frac{1}{K}\sum_{k=1}^{K} r_{i,k}, \qquad \sigma_i^2 = \frac{1}{K}\sum_{k=1}^{K}(r_{i,k} - \mu_i)^2. \tag{28}$$

We clip the centered response-level advantage:

$$A_{i,k} = \mathrm{clip}\left(A^{\mathrm{resp}}_{i,k}, -c,\, c\right), \tag{29}$$

Prompt-level scores such as $b(h_i)$ or trajectory-level annotations may be used to filter or stratify prompts before sampling, but they are not shifted into $A_{i,k}$; this keeps the advantages centered within each prompt group and avoids introducing a constant-gradient bias.

Policy updates use a clipped policy-gradient objective regularized by a reference model. Let

$$\rho_{i,k,t} = \frac{\pi_\phi(a_{i,k,t} \mid p_i, a_{i,k,<t})}{\pi_{\mathrm{old}}(a_{i,k,t} \mid p_i, a_{i,k,<t})}. \tag{30}$$

We optimize

$$\mathcal{L}_{\mathrm{RL}} = -\mathbb{E}\left[\min\Big(\rho A,\ \mathrm{clip}(\rho, 1-\epsilon, 1+\epsilon)A\Big)\right] + \eta\, \mathcal{D}_{\mathrm{ref}}(\pi_\phi, \pi_{\mathrm{ref}}), \tag{31}$$

where $\mathcal{D}_{\mathrm{ref}}$ is a reference-model regularizer. Since the reward used in Eq. (31) is computed by the frozen mental-state and reward model, the policy is trained *under* the learned Theory-of-Mind model, even though the latent state is not fed explicitly to the policy at deployment. The implementation supports gradient accumulation, reward clipping, cosine learning-rate scheduling, and checkpoint selection based on rolling validation reward.

A.1.6 Deployment and Evaluation

At deployment time, only the learned policy is retained: the reward model is not used for test-time search or reranking. For open-ended interaction spaces, the policy directly generates a natural-language response. For structured action spaces, the same wrapper prompts the policy to emit a format-constrained output and parses it into the benchmark's expected action object. Episodes are then executed in each benchmark's native environment with the trained policy occupying one agent slot and the fixed partner or evaluator specified by the benchmark protocol.

A.2 Dataset-Specific Instantiations

The shared objective is defined once in Appendix A.1.4. To separate methodological choices from dataset bookkeeping, each subsection below follows the same order: data construction, training instance, and policy/evaluation protocol. We specify only the context, utility target, mental-state supervision, and contrastive signal that change across datasets rather than re-expanding the same loss terms.

A.2.1 BIGTOM

Original dataset. BIGTOM [7] is a procedurally generated theory-of-mind benchmark of 200 scenarios. Each scenario is a five-sentence story with context, desire, perception cue, belief, and an external causal event that changes the object's state, paired with three questions (belief, desire, action) and gold answers for both branches: the agent *is aware* of the causal event, or *is not aware*.

Augmentation. We expand the seed set 25× by prompting Qwen3-235B with $k{=}4$ randomly sampled seeds as in-context exemplars and the original 19-field BIGTOM schema. We sample 5,000 new scenarios at temperature 1.0 and retain only those passing a schema validator: exactly five sentences, the fourth beginning with a belief statement, distinct aware/not-aware branches, and a random event disjoint from the causal event. Generated scenarios are deduplicated against prior outputs and preserved the original BIGTOM schema, allowing drop-in compatibility with downstream tooling.

Because BIGTOM labels only first-order beliefs, we enrich each scenario with a second-order belief by prompting Qwen3-235B in reasoning mode (temperature 0.4) to produce: (i) an invented *observer*, a plausible second character physically present at the event; (ii) a paraphrase of the story; and (iii) for each of the aware and not-aware conditions, a first-order belief (the agent's belief about the object), a second-order belief (the agent's belief about what the observer believes about the object), and a 2–4 sentence chain-of-thought rationale ending in the predicted action. A validator rejects outputs with empty fields or identical first-order beliefs across conditions.

Resulting training set. Each annotated synthetic scenario is flattened into two training examples, one per perceptual condition, yielding 9,964 rows from 5K generated BIGTOM-style scenarios. Every row carries BIGTOM-compatible gold labels (`gold_belief`, `gold_action`) alongside our added fields (`first_order_belief`, `second_order_belief`, `rationale`, `observer_name`, `observer_role`, `paraphrase`). This synthetic set is used for training; the official BIGTOM benchmark and TOMI are held out for evaluation. The rationale serves as the supervision target for mental-state reasoning, and the second-order belief enables training and evaluation of higher-order theory-of-mind beyond the scope of the original benchmark.

Training instance. For a scenario s, perceptual branch $b \in \{\text{aware}, \text{not_aware}\}$, initial-belief flag ι, and question q, the context is $c_i^{\text{BT}} = [\text{story}_\iota(s); q]$. The positive answer $a_i^+ = y(s, b, q)$ is paired with the opposite-branch answer $a_i^- = y(s, \bar{b}, q)$ as a hard negative. The scalar utility target $u_i^{\text{BT}} \in \{0, 1\}$ denotes branch-consistent correctness, so this setting uses scalar reward regression and an unmasked preference term. Generated first- and second-order beliefs supervise mental-state reconstruction, while the rationale is included in the mental-reasoning target. The branch label is used only by the optional aware/not-aware diagnostic classifier described above.

Optimization and evaluation. Policy learning uses mental-prefix SFT on the same QA prompts followed by GRPO, with the frozen scalar action-conditioned utility head providing the response reward and a reference-policy KL penalty. Prompt-level latent-only predictions are retained for diagnostics rather than added to the group-relative advantage. Evaluation uses the official true-belief, false-belief, and paired TB∧FB metrics.

A.2.2 SOTOPIA

Original dataset. SOTOPIA-π [30] builds on the SOTOPIA environment [35], which consists of 90 social scenarios and 40 character profiles. Each episode pairs two characters in a scenario and assigns each a private social goal, covering negotiation, collaboration, and competition. Character profiles include name, gender, occupation, personality description, and secrets. The released dataset contains multi-turn social interaction episodes, each with a full conversation and episode-level scores across seven social dimensions: believability, relationship, knowledge, secret-keeping, social rules, financial benefits, and goal completion.

Augmentation. We apply a two-stage GPT-4o annotation pipeline to each episode, decomposing each episode turn by turn and annotating every turn for both agents:

Stage A: reward scores with reasoning across all seven social dimensions, and first-order mental state annotations – `partner_belief` (what the agent believes about the partner's hidden intentions), `strategic_intent` (the agent's sub-goal for this specific turn), `thought_process` (the latent reasoning connecting belief to action), and `hard_negative_response` (a fluent but socially poor alternative response for contrastive learning).

Stage B: adds second-order theory-of-mind annotations for each agent-turn – `second_order_belief` (what agent B thinks agent A believes), `second_order_intent`

(what agent B thinks agent A intends), and `second_order_thought` (what agent B thinks agent A is thinking).

Resulting training set. In addition to the original SOTOPIA-π conversation episode context (scenario, agent profiles, goals, secrets, and conversation), we generated per-turn reward scores with reasoning across all seven dimensions for both agents, alongside first-order and second-order mental state annotations, yielding 1,647 fully annotated episodes. These augmented SOTOPIA-π episodes are used for training the mental-state/reward model; official SOTOPIA evaluation episodes are held out.

Training instance. For a SOTOPIA turn t in episode e, the context includes the public scenario, both character profiles, private goals/secrets for the acting agent, and the dialogue prefix:

$$c_{e,t}^{\mathrm{ST}} = \big[s_e;\ \rho_A; \rho_B;\ g_A; g_B;\ \mathrm{secret}_A; \mathrm{secret}_B;\ h_{e,<t}\big]. \tag{32}$$

The positive action is the observed socially grounded utterance $a_{e,t}^{+}$, while $a_{e,t}^{-}$ is a generated hard-negative response that remains fluent but violates a social, strategic, or private-goal constraint. The utility vector is seven-dimensional,

$$\mathbf{u}_{e,t}^{\mathrm{ST}} = \big[u_{\mathrm{bel}}, u_{\mathrm{rel}}, u_{\mathrm{know}}, u_{\mathrm{secret}}, u_{\mathrm{rules}}, u_{\mathrm{fin}}, u_{\mathrm{goal}}\big] \in [0,1]^7, \tag{33}$$

obtained by normalizing the SOTOPIA dimensions from their official score ranges. The first-order mental target concatenates partner belief, strategic intent, and thought process; the second-order target concatenates second-order belief, second-order intent, and second-order thought.

Optimization and evaluation. SOTOPIA instantiates all four supervised terms in Eq. (18): utility regression fits the seven-dimensional score, mental reconstruction decodes the two concatenated ToM targets, rationale conditioning predicts the same utility vector, and preference learning contrasts the observed response with the generated negative. The mask $\gamma_{e,t,d}^{\mathrm{ST}}$ activates the preference penalty only for dimensions intentionally degraded by that negative, avoiding penalties on qualities such as fluency or believability when they remain valid. Latent-only utility and future-action prediction are used only as the optional regularizers summarized above. GRPO scalarizes the seven predicted dimensions with $w_{\mathrm{ST}} \in \Delta^6$, and the deployed policy is evaluated with the official SOTOPIA evaluator on those dimensions and their mean.

A.2.3 MMROLE

Original dataset. MMROLE [4] is a multimodal role-playing benchmark consisting of 85 characters, 11K images, and 14K single or multi-turn dialogues, yielding 85K training samples. Characters span fictional figures (Marvel heroes, Harry Potter, Friends), real historical persons (Kobe Bryant, Leonardo da Vinci), and Chinese historical figures. Each episode pairs a character profile (including personality, life story, relationships, and catchphrases) with a real image from COCO or movie/TV scripts. Three dialogue types are covered: *comment* (a character comments on an image), *human-role* (a human and a character discuss an image), and *inter-role* (two characters interact about an image). The original dataset evaluates role-playing quality across eight dimensions: coherency, fluency, image-text relevance, instruction adherence, knowledge consistency, personality consistency, response accuracy, and tone consistency.

Augmentation. We focus exclusively on *inter-role* dialogues and apply a three-step pipeline to add visual theory-of-mind annotations.

Step 1 – Per-turn restructuring: Each inter-role dialogue is decomposed into per-turn snapshots. Each turn contains the speaker and partner profiles, the image, the full dialogue history up to that turn, and the current utterance, producing 36,744 per-turn examples.

Step 2 – Visual ToM annotation: Each per-turn example is annotated by GPT-4o with access to the actual image, producing structured belief states:

- **Visual percepts**: objects in the scene and their salience (high/medium/low/none) for each character *based on their personality* — a nurse and a politician attend to different elements of the same image.
- **First-order beliefs**: what the speaker believes the partner is visually attending to and intending.

- **Second-order beliefs**: what the speaker believes the partner believes about the speaker.
- **Belief probes**: structured question-answer pairs across five types (visual percept, first-order belief, second-order belief, intent inference, false belief) at three difficulty levels.
- **Contrastive belief states**: a ToM-violating alternative response for preference learning.

Step 3 – Validation: Annotations are quality-filtered across six checks — object count, salience asymmetry between agents, belief-scene grounding, probe answer distinctiveness, contrastive response distinctiveness, and second-order belief consistency — producing a clean set of 15,607 annotated turns.

Resulting training set. The 15,607 annotated turns are converted into four training formats using a character-stratified split (74 train characters, 14 held-out test characters):

- **Belief prediction** (12,762 train rows): given image, agent profiles, and dialogue history, predict the full structured belief state.
- **Preference pairs** (50,952 train rows): ToM-aligned responses paired against four types of violation, wrong visual attribution, wrong first-order belief, wrong second-order belief, and no theory-of-mind baseline, yielding four contrastive pairs per example.
- **Probe QA** (63,810 train rows): direct question-answer pairs for ToM evaluation and reward model fine-tuning.
- **Salience prediction** (25,524 train rows): given image and one character profile, predict per-object salience ratings.

Thus, MMROLE uses the released training dialogues as the base resource, augments them with visual first-order/second-order mental labels and contrastive ToM violations, and evaluates on the held-out role split under the official MMROLE protocol.

Training instance. For MMROLE, the context is multimodal:

$$c_i^{\mathrm{MM}} = \big[I_i;\ \rho_i^{\mathrm{speaker}}; \rho_i^{\mathrm{partner}};\ h_{i,<t};\ \mathrm{role_instruction}_i\ \big], \tag{34}$$

where I_i is the image and ρ denotes the character profile. A vision-language backbone jointly encodes the image and textual context, and its pooled representation parameterizes the two latent posteriors. The positive response is the current in-character utterance, while negatives are contrastive ToM violations that alter visual grounding, first-order belief, second-order belief, or remove ToM reasoning. For each negative, $\gamma_{i,d}^{\mathrm{MM}}$ marks which official dimensions the contrastive edit is intended to violate. We obtain $\mathbf{u}_{i,-}^{\mathrm{MM}}$ by applying the same official MMROLE evaluator and normalization to the contrastive negative response under the unchanged image, profile, and dialogue context. If a particular ablation does not score negatives with the evaluator, this regression term is masked. The joint reward vector is the official eight-dimensional MMROLE score:

$$\mathbf{u}_i^{\mathrm{MM}} = \big[u_{\mathrm{IA}}, u_{\mathrm{Flu}}, u_{\mathrm{Coh}}, u_{\mathrm{ITR}}, u_{\mathrm{RA}}, u_{\mathrm{PC}}, u_{\mathrm{KC}}, u_{\mathrm{TC}}\big] \in [0,1]^8. \tag{35}$$

For diagnostics, we also derive an optional three-dimensional latent-only target covering first-order belief, second-order belief, and belief divergence; this is one of the lightweight probe regressors summarized in Appendix A.1.4 rather than a core term.

Optimization and evaluation. The multimodal reward model uses the shared utility-regression, preference, mental-reconstruction, and KL terms. Utility regression fits the official eight-dimensional score, visual mental reconstruction decodes first- and second-order targets conditioned on the image, and $\gamma_{i,d}^{\mathrm{MM}}$ restricts each preference comparison to dimensions changed by that contrastive edit. When an official score for a negative response is available, we optionally regress to it directly; the three-dimensional latent-only ToM target and future-action prediction are likewise auxiliary regularizers rather than defining components of the method.

Policy training then uses visual mental-prefix SFT, GRPO with the frozen multimodal reward vector scalarized by $w_{\mathrm{MM}} \in \Delta^7$, and standard DPO on the contrastive preference pairs. Evaluation follows the official MMROLE protocol and reports the normalized eight dimensions and their overall score.

Appendix Table 10 summarizes which supervision fields are present in the original releases versus our augmented training resources. In terms of scale, our training data consists of 1,647 annotated SOTOPIA-π episodes, 5K synthetic BIGTOM-style scenarios flattened to 9,964 rows, and 15,607 MMROLE inter-role turns converted into belief-prediction, preference, probe-QA, and salience-training formats. Official BIGTOM, TOMI, SOTOPIA, MMROLE held-out roles, CRAIGSLIST-BARGAIN, and FANTOM are used as evaluation or transfer settings rather than as training data.

Table 10: Comparison of original and augmented datasets across annotation dimensions.

Dataset	Multi-modal	Episode Reward	Per-turn Reward	Intent Annot.	1st-order Belief	2nd-order Belief	Rationale / CoT	Hard Negatives
BIGTOM (original)	✗	✗	✗	✗	✓	✗	✗	✗
BIGTOM (ours)	✗	✗	✗	✓	✓	✓	✓	✓
SOTOPIA-π (original)	✗	✓	✗	✗	✗	✗	✗	✗
SOTOPIA-π (ours)	✗	✓	✓	✓	✓	✓	✓	✓
MMROLE (original)	✓	✓	✗	✗	✗	✗	✗	✗
MMROLE (ours)	✓	✓	✓	✓	✓	✓	✗	✓

A.2.4 Mental-State Annotation and Human Verification

Mental-state annotations are generated from the source evidence available in each benchmark, including the scenario or image, character profiles, dialogue history, and current turn. Before training, every supervision set passes automated structural and semantic-consistency checks. On MMROLE, the six-check validator retained $15{,}607/15{,}919$ training turns (98.04%) and $291/294$ examples from the official test partition.

We additionally conducted a blinded human audit of 100 randomly sampled annotations (50 each from MMROLE and SOTOPIA). Twelve evaluators saw the underlying evidence and generated annotation, but not the dataset or generator identity, and rated whether each mental-state claim was accurate and supported on a five-point scale. The overall mean was $\mathbf{4.45/5}$ (89.0% after linear rescaling), with $90/100$ annotations receiving at least 4 and 56 receiving 5. The lower-rated cases primarily involved overconfident or unsupported inferences, underspecified claims, or contradictions with the dialogue or image. Because the audit prioritizes coverage of distinct records rather than repeated ratings per record, we treat it as a blinded quality spot-check rather than an inter-rater agreement study.

Table 11: Blinded human audit of generated mental-state annotations. Faithfulness linearly rescales the mean rating to 0–100%.

Subset	n	Mean (1–5)	Faithfulness (%)
MMROLE	50	4.30	86.0
SOTOPIA	50	4.60	92.0
Overall	**100**	**4.45**	**89.0**

A.3 Additional Results and Ablations

Zero-Shot Multi-Party Transfer to FANTOM. We test whether a mental-model policy trained only on BIGTOM-style data transfers beyond dyadic interaction. Without any FANTOM training, fine-tuning, or adaptation, we evaluate Base, SFT, and our policy on the first $5{,}000$ flattened probes of FANTOM [14], a multi-party, information-asymmetric ToM benchmark, using its official evaluation protocol and scorer. Our policy improves over Base on all four metrics and yields the strongest second-order score (52.2 vs. 49.7 for Base and 41.4 for SFT), providing preliminary evidence that the learned mental-model signal transfers to multi-party reasoning.

BIGTOM Latent Geometry. The unsupervised UMAP in Figure 6 provides a view of the BIGTOM latent geometry. Task identity forms the dominant macro-structure, while aware and not-aware

Table 12: Zero-shot BIGTOM→FANTOM transfer on 5,000 probes using the official scorer. Scores are percentages; higher is better.

Model	Belief choice	Belief distance	First order	Second order
Base	25.4	35.2	26.8	49.7
SFT	26.5	33.8	29.4	41.4
Ours	**27.9**	**38.0**	**29.7**	**52.2**

branches remain locally separated inside nearby task/scenario regions. This supports the probe results in the main analysis: belief state is not the largest visual axis of variation, but it is still represented in a recoverable and geometrically coherent direction.

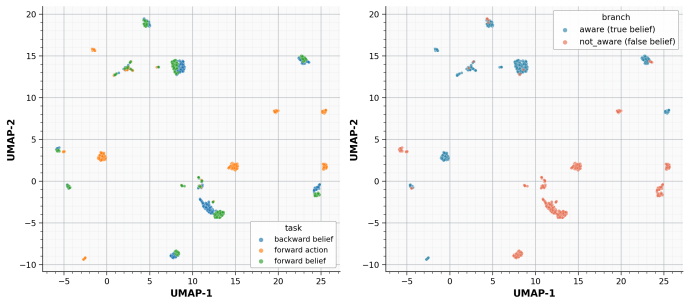


Figure 6: Mental-latent geometry from an unsupervised UMAP of latent z. **Left:** coloring by task shows task/scenario macro-clusters. **Right:** coloring by branch shows aware and not-aware samples remain near paired scenarios but are displaced within task regions, consistent with branch F1 $= 0.989$.

Table 13: BIGTOM official-protocol results. TB denotes true-belief accuracy, FB denotes false-belief accuracy, and TB∧FB denotes paired accuracy, where both the true-belief and false-belief versions of the same scenario must be answered correctly.

		Without Initial Belief			With Initial Belief		
Task	Model	TB	FB	TB∧FB	TB	FB	TB∧FB
Forward Belief	GPT-4 (0-shot)	99.0	98.0	97.0	91.0	99.0	90.0
	GPT-4 (0-shot-CoT)	99.0	99.0	98.0	99.0	99.0	98.0
	GPT-4 (1-shot)	99.0	99.0	97.0	97.0	99.0	96.0
	GPT-4 (1-shot-CoT)	100.0	99.0	99.0	97.0	99.0	96.0
	Base	83.0	94.0	77.0	49.5	98.5	48.0
	SFT	95.5	88.5	84.5	94.0	96.5	92.0
	Ours	**97.0**	**98.0**	**96.0**	**97.5**	**99.5**	**97.5**
Forward Action	GPT-4 (0-shot)	98.0	81.0	79.0	98.0	92.0	90.0
	GPT-4 (0-shot-CoT)	99.0	88.0	87.0	99.0	96.0	95.0
	GPT-4 (1-shot)	100.0	98.0	98.0	100.0	100.0	100.0
	GPT-4 (1-shot-CoT)	100.0	100.0	100.0	100.0	99.0	99.0
	Base	99.5	64.0	63.5	**99.5**	68.0	67.5
	SFT	93.0	82.5	76.5	92.0	80.0	73.0
	Ours	**99.5**	**79.0**	**78.5**	99.0	**81.0**	**80.0**
Backward Belief	GPT-4 (0-shot)	86.0	53.0	40.0	62.0	77.0	40.0
	GPT-4 (0-shot-CoT)	84.0	54.0	38.0	76.0	63.0	40.0
	GPT-4 (1-shot)	68.0	85.0	53.0	57.0	92.0	49.0
	GPT-4 (1-shot-CoT)	83.0	75.0	58.0	81.0	85.0	65.0
	Base	85.0	92.5	77.5	53.5	99.0	52.5
	SFT	94.5	90.5	85.0	94.5	95.5	91.0
	Ours	**97.0**	**98.5**	**95.5**	**98.5**	**99.5**	**98.5**

BIGTOM official-protocol comparison. Table 13 expands the main BIGTOM result with the full official prompting protocol, including zero-shot, CoT, and one-shot GPT-4 references. The most diagnostic columns are TB∧FB, because they require the model to answer both true- and false-belief branches of the same scenario correctly rather than solving only the easier branch. Our model gives the strongest open-source paired accuracy in forward belief and backward belief, and improves the

paired forward-action scores over the base model in both settings. This pattern suggests that the mental-state reward signal mainly helps with counterfactual consistency: the policy is less likely to collapse to the observable world state when the initial belief sentence or the false-belief branch changes.

First- versus second-order mental state. Figure 7 visualizes the comparison of Qwen2.5-7B variant for first-order and second-order mental-state latents on the **SOTOPIA** benchmark. Each petal shows the second-order change as a percentage of the official range for that SOTOPIA dimension, while the center badge reports the relative lift in the average score. The second-order latent gives a broader outward profile, especially on task-facing dimensions such as knowledge, goal completion, financial reasoning, and relationship management, suggesting that recursive partner modeling adds useful signal beyond a first-order mental state alone.

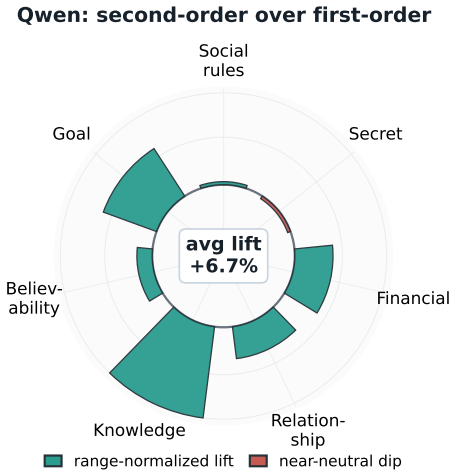


Figure 7: Range-normalized lift from Qwen2.5-7B (ours w/ 1st order) to Qwen2.5-7B (ours w/ 2nd order) on SOTOPIA. Petal lengths encode percentage-of-range changes for each official dimension rather than raw scores. The center reports the relative lift in the average score, highlighting that second-order ToM improves the aggregate profile while keeping the comparison scale-free.

BIT Factorization Sharpens Preference Margins. We ablate the BIT decomposition by retraining identical encoders with two control supervision signals: a *flat summary*, where belief, intent, and thought are concatenated into a single free-form target decoded from the full latent, and a *shuffled* variant, where the three labels are randomly reassigned to slots before training. All three variants reach near-ceiling preference accuracy on the SOTOPIA validation set. The preference loss, however, separates them by two orders of magnitude (BIT 9.6×10^{-5} vs. flat 1.0×10^{-2}), because $\mathrm{softplus}(r^- - r^+)$ measures the margin by which positives are ranked above negatives, not only whether the ranking is correct. Figure 8 shows the per-sample margin $r^+ - r^-$: BIT is centered at $+25.06$ median, with every sample correctly ranked at margin $\geq +1.40$, while flat summary is centered at $+12.76$ and overlaps the shuffled-label control ($+12.53$ median). This suggests that mental content alone is not enough; routing belief, intent, and thought to dedicated latent sub-blocks is what yields confident partner-conditioned preference separation.

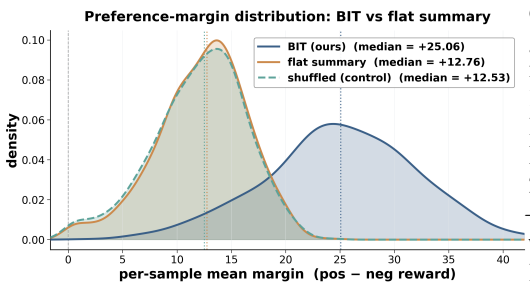


Figure 8: **Preference-margin distribution.** BIT decomposition produces sharper preference margins than flat or shuffled mental supervision. Kernel density of the per-sample margin $r^+ - r^-$ on 1,508 SOTOPIA validation turns. Dotted vertical lines mark medians; the dashed line marks zero. BIT ranks positives over negatives with roughly $2\times$ higher confidence (median $+25.06$ vs. flat $+12.76$), and every BIT sample is correctly ranked with margin $\geq +1.40$. Flat overlaps the shuffled-label control, indicating that mental supervision improves calibration when belief, intent, and thought are routed to dedicated latent sub-blocks.

Cross-agent mental-reward transfer. Figure 9 compares the aggregate SOTOPIA score under the two reward models. The transferred Qwen mental-reward model does not reduce performance and slightly improves the average score (3.557 vs. 3.541), indicating that the learned mental-reward architecture is not tied to a single policy family and can provide a useful scoring signal for another agent.

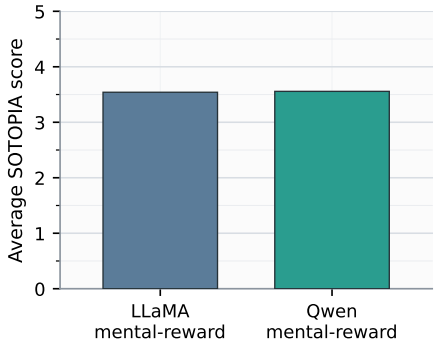


Figure 9: Cross-agent transfer of the mental-reward model on SOTOPIA. Bars compare the average scores of LLaMA2-7B (ours) with its same-architecture mental-reward model and LLaMA2-7B (ours w/ r-Qwen) with the Qwen mental-reward model. The transferred Qwen mental-reward model preserves overall performance, supporting its generalization across agents.

Role-Addressable Latent Blocks. We additionally perform a role-swap intervention in latent space (Fig. 10). Although the absolute effects are small, BIT is the only variant where swapping a role-specific latent block produces a greater semantic pull on the matching role prediction than on non-matching roles; Flat and Shuffled remain diffuse. This suggests that the BIT decomposition induces weak but measurable role-addressability.

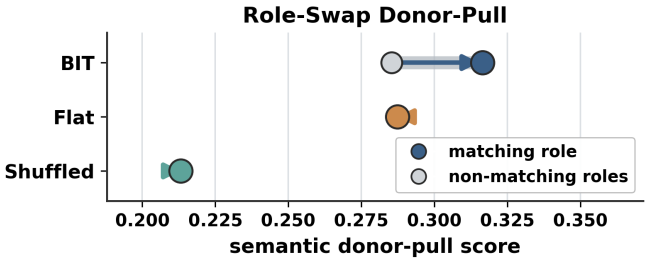


Figure 10: **Role-swap latent intervention.** Swapping a BIT role-specific block moves the corresponding role prediction more than unrelated roles. Flat and Shuffled variants do not show this separation, yielding more diffuse effects.

Interpreting z-space and Ablation on Emotion Structure in Dialogue-Level Mental Latents We conduct an ablation study to test whether the z-space (mental-state latent space) learned by the recursive ToM model captures affective information beyond the model's original SOTOPIA training objective. We use a dialogue-level DailyDialog [17] dataset subset filtered to contain only dialogues with a single consistent non-neutral emotion, ignoring neutral turns and merging anger/disgust due to their strong overlap. This yields a balanced 5-way emotion classification setting. Each dialogue is represented in a structured dialogue history format with turn indices and speaker markers, and we append the prompt suffix *Emotion:* so that the final-token latent is aligned with an emotion-prediction query. We compare z-probe representations against no-adapter Qwen baseline, to test whether emotion separability comes from the learned mental latents rather than from the base language model alone. We assess the embeddings along two complementary axes:

1. 5-fold cross-validated 5-NN accuracy and silhouette score, both computed on the standardized high-dimensional embeddings. The 5-NN accuracy measures how easily emotion labels can be inferred from the embedding geometry, while the silhouette score measures class compactness and separation in the original high-dimensional space.

2. t-SNE visualization at perplexity 15 Figure 11, which projects each embedding to two dimensions for qualitative inspection of class structure. It provides a complementary geometric view of how the classes are arranged.

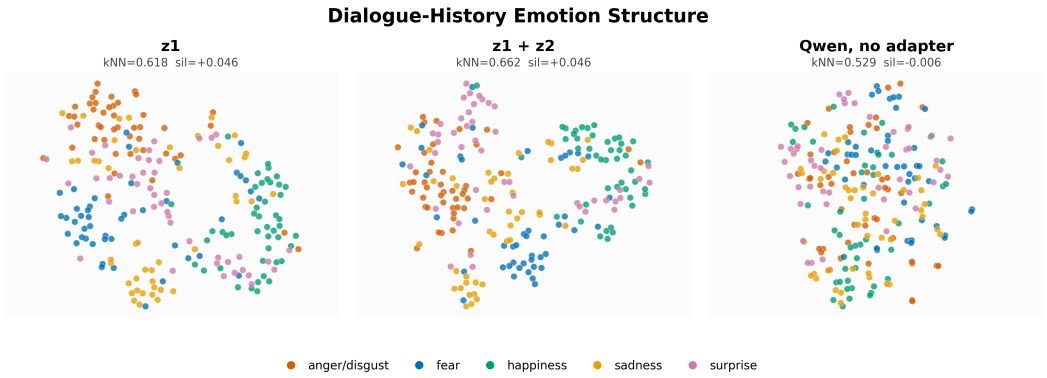


Figure 11: **t-SNE visualization** The concatenated latent representation [z1; z2] reaches 0.662 5-NN accuracy (silhouette +0.046) on the 5-way emotion task compared to 0.529 for Qwen without the adapter, and a chance baseline of 0.200. The +0.13 gap over the no-adapter baseline indicates that the learned mental latents encode emotion-relevant structure that is not present in the base Qwen representation. The corresponding t-SNE projection provides a qualitative visualization: classes appear more separated in the z1 and [z1; z2] projections, while the no-adapter Qwen projection shows more inter-class overlap.

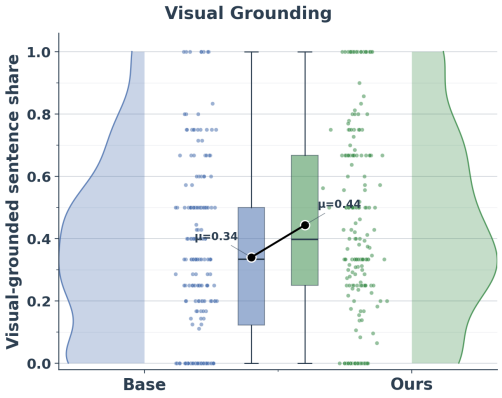


Figure 12: **Visual Grounding.** Visual grounding rate per response for Base vs. Ours on 200 held-out MMROLE examples. Each point is one example's share of generated sentences that mention a gold scene object; black markers show means. The metric is computed using a token-overlap classifier between generated sentence tokens and the gold `scene_objects` token bag, making it a proxy for grounding rather than grounding itself. Ours increases the rate by $\Delta = +0.103$ (paired Wilcoxon $p = 1.3 \times 10^{-4}$, ***).

Visual Grounding in MMROLE. Figure 12 provides an appendix-level check that the MMROLE gains are accompanied by stronger use of visual evidence. We measure a response-level visual-grounding proxy on 200 held-out examples by splitting each generated response into sentences and computing the share of sentences that mention a gold scene object. The detector is an operational token-overlap classifier between generated sentence tokens and the gold `scene_objects` token bag, so it should be read as a proxy for grounding rather than grounding itself. Under this metric, ours increases the mean visual grounding rate over the base model by 0.103 with a paired Wilcoxon $p = 1.3 \times 10^{-4}$, suggesting that mental-model training keeps generated role responses more tied to objects present in the scene.

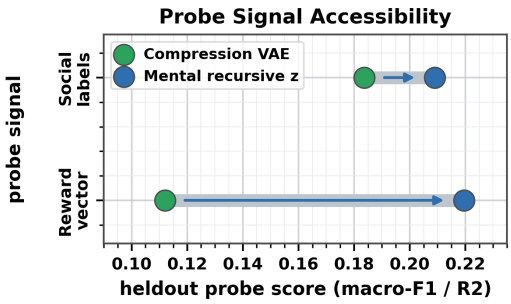


Figure 13: **Probe signal accessibility.** We compare the recursive mental-model latent against a compression-only VAE trained on the same SOTOPIA-π contexts and responses with matched latent dimensionality. The VAE is optimized only for reconstruction, while the mental-model encoder is trained with ToM supervision and reward coupling. After freezing both encoders, we fit simple probes from z and evaluate held-out recovery of social-label structure and reward-vector targets. Higher macro-F1/R^2 for the mental latent shows that downstream social and reward signals are more accessible than in a generic compression code.

Mental-model is not a compression code. To test whether the mental-model latent is more than a compression code, we compare it against a compression-only VAE trained on the same SOTOPIA-π contexts and responses, using the same latent dimensionality and a reconstruction objective. We freeze both encoders, train lightweight linear probes from the latent to task-relevant targets, and evaluate recovery of social-label structure and reward-vector dimensions on held-out examples. Figure 13 shows that the probes recover both targets more effectively from the mental latent than from the compression VAE. This indicates that the recursive mental-model objective does not merely compress the context; it organizes the latent space around downstream social and reward signals.

A.4 Impact Statement

This work studies explicit mental-model representations for multi-agent systems. The main positive impact is to make agent behavior more interpretable and controllable in settings where an agent must reason about another agent's goals, beliefs, knowledge, or visual perspective. Such representations can support safer assistants, better negotiation and collaboration agents, and clearer diagnostic tools for evaluating whether a model is relying on task-relevant social and perceptual signals rather than surface imitation. Our augmented datasets also provide structured supervision for studying Theory-of-Mind behavior across language-only and multimodal benchmarks.

The same capability also carries risks. Models that infer or exploit another agent's latent state could be misused for manipulation, persuasion, privacy invasion, or profiling, especially if deployed in high-stakes human-facing settings. The inferred variables in our framework should therefore be treated as task-conditioned modeling constructs, not as ground-truth psychological states. They may reflect annotation bias, dataset artifacts, or stereotypes in the underlying models and data. For this reason, deployment should include consent-aware data collection, privacy filtering, bias audits, uncertainty reporting, and restrictions against using inferred mental states for sensitive decisions about individuals. Our experiments are limited to benchmark and synthetic/augmented training settings, and future work should study human oversight and safety constraints before using such systems in real-world multi-agent applications.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Refer to section 4

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

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

   Answer: [Yes]

   Justification: Refer to section 6

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: The paper does not include theoretical results

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Refer to sections 4, A.2, and A.1.3

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will release the code, models and data upon acceptance.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Refer to sections 4,A.2 and A.1.3

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The paper reports official benchmark scores under fixed evaluation protocols and includes a paired Wilcoxon significance test for the MMROLE visual-grounding analysis in Appendix 12. For SOTOPIA, we repeated the main evaluations multiple times and observed nearly identical aggregate scores, suggesting low evaluation variance under the fixed protocol. However, we do not report formal run-to-run error bars or confidence intervals for all main trained-policy comparisons, since repeating full 7B-scale LLM/VLM training and closed-source evaluations across multiple seeds is computationally and API-cost intensive. We therefore treat the main benchmark tables as point estimates and support the claims through consistency across benchmarks, model families, held-out splits, repeated SOTOPIA evaluations, and ablations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Refer to sections 4, A.1.3 and A.2

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes paper conforms with NeurIPS code of ethics

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Refer to section A.4

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: All the models and data used in the paper are publicly available.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Refer to sections 1, 2 and 4

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Refer to sections 4, A.1.3 and A.2

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve any experiments with humans

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: We did not use LLM for any method specific ideas

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.