Know You Before You Speak: User World Modeling for LLM Personalization in Multi-Turn Conversation

Jiani Luo[1], **Xiaoyan Zhao**[1*], **Yang Zhang**[1*], **Shuyi Miao**[2]
Bingbing Xu[3], **Stefan Konigorski**[4], **Tat-Seng Chua**[1]

[1]School of Computing, National University of Singapore, Singapore
[2]School of Artificial Intelligence, Beihang University, Beijing, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[4]German Institute of Human Nutrition Potsdam-Rehbruecke, Germany
xzhao@se.cuhk.edu.hk, zhangy@nus.edu.sg, e1616036@u.nus.edu

Abstract

Personalized dialogue requires more than recalling explicit user histories: systems also need to infer hidden user states that evolve through interaction and shape appropriate response strategies. Existing memory- and profile-based methods primarily reuse observable user information, offering limited support for modeling user-state dynamics or selecting actions based on how they shape future user states. We propose **PUMA** (**P**rospective **U**ser world **M**odeling for **A**ction Selection), a framework grounded in the Free Energy Principle (FEP) that formulates personalization as decision-making under partial observability, centered on an explicit ***user world model*** that captures latent user states and their action-conditioned dynamics. At each turn, PUMA maintains a belief over the user's hidden state, refines the user world model for observation generation and action-conditioned state transition, and selects dialogue actions by minimizing expected free energy—balancing epistemic and pragmatic objectives under a unified criterion. This formulation shifts personalization from passive memory retrieval to model-based decision-making over user evolution. We instantiate PUMA on healthcare-oriented counseling and motivational interviewing benchmarks with latent state annotations for rigorous evaluation. Experiments show that PUMA improves long-horizon dialogue outcomes while maintaining strong response quality, and a cross-dataset study demonstrates more reliable user-state estimation and next-state prediction. Our code is available at: https://github.com/Annie1161/PUMA.

1 Introduction

Personalized dialogue systems aim to generate responses that remain coherent, adaptive, and user-specific over long-term interactions [56, 16, 2]. They are important in applications such as personal agents/assistants, education, healthcare communication, mental health support, and long-term decision support, where effective interaction depends on understanding both the current query and the user's evolving context. Recent LLM-based systems [57, 28, 55, 38, 52, 45] have advanced personalization by incorporating long-term memory, user profiles, and retrieved interaction histories, enabling models to reuse explicit user information such as preferences, decisions, and prior dialogue content.

However, long-term personalization cannot be reduced to recalling explicitly stated user information. The same utterance may require fundamentally different responses depending on the user's latent state [3]. Consider a user who says, "I've been taking the new medication for two weeks". The

*Corresponding authors.

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

appropriate response depends on whether the user is anxious about side effects, confidently tracking progress, or subtly reconsidering discontinuation. These differences reflect latent factors such as uncertainty, engagement, and readiness to act [14], which are only partially expressed in language but critically shape how an utterance should be interpreted and how the system should respond [3, 36, 19, 22]. Therefore, effective personalization requires not only tracking what the user has said, but also inferring hidden user states and reasoning about how system actions influence their evolution over time. Such reasoning requires a predictive account of both how latent states give rise to observable utterances and how they change in response to system actions. This points to the need for a **user world model**—a predictive representation of the user as a partially observable dynamical system—as the foundation for long-term personalization.

Existing personalized LLM-based systems remain limited from this perspective. Memory-augmented and profile-based methods [54, 13, 48, 39] operate over explicit textual records, making them effective for recalling stable facts but insufficient for modeling latent state evolution. While some approaches introduce latent variables, these representations are typically static or descriptive, failing to capture the continuous and action-dependent dynamics of user behavior [41, 28, 17]. As a result, existing systems implicitly treat personalization as a retrieval or conditioning problem, rather than a sequential decision-making problem over a partially observable user. In short, current methods either lack a user world model entirely or rely on limited surrogates that cannot predict how users will respond to system actions—leaving prospective reasoning about user evolution out of reach.

The Free Energy Principle (FEP) and active inference [6, 8] provide a principled decision-theoretic foundation for addressing this gap. Under FEP, an agent operates in a partially observable environment by maintaining and updating beliefs over hidden states, and selecting actions by minimizing expected free energy (EFE), which balances uncertainty reduction (epistemic value) and goal-directed behavior (pragmatic value). Since EFE is evaluated over action-conditioned dynamics, actions are chosen based on both their immediate effects and their influence on future state beliefs. This formulation naturally aligns with personalized dialogue, where the user is a latent dynamical system and system responses act as interventions that shape its evolution.

Building on this perspective, we propose **PUMA** (**P**rospective **U**ser world **M**odeling for **A**ction selection), an FEP-grounded framework for personalization. At the core of PUMA is a *user world model, a predictive model over user observations, latent user states, and their action-conditioned dynamics that captures how the user evolves in response to system interventions*. This goes beyond static state estimation by explicitly modeling state transitions induced by dialogue actions. At each dialogue turn, PUMA updates its belief over the latent user state under the user world model, refines the model's transition and observation components, and evaluates candidate dialogue actions through an approximation of expected free energy. This enables joint inference and control within a unified decision framework grounded in the FEP. This formulation shifts personalization from passive memory retrieval to model-based decision-making over user evolution. Rather than selecting responses based on retrieved context or static preferences, PUMA evaluates candidate actions by their expected influence on future user trajectories, balancing epistemic and pragmatic objectives under a single principled criterion.

We instantiate PUMA on healthcare dialogue benchmarks, which provide the latent user state annotations necessary for evaluating belief tracking and long-horizon decision quality. Experimental results show that explicitly modeling user state dynamics and performing FEP-based action evaluation significantly improve both user state tracking and long-term performance compared with strong LLM-based baselines.

Our main contributions are summarized as follows:

- We introduce a user world model for personalized dialogue that, unlike static persona, memory-based or profile-based representations in prior work, captures latent user states and their action-conditioned evolution over time.
- We formulate personalized dialogue under the Free Energy Principle and develop PUMA, an FEP-grounded framework that unifies belief updating and action selection via expected free energy, enabling prospective decision-making by reasoning over future user states.
- We empirically evaluate PUMA on healthcare dialogue benchmarks, where FEP-grounded user world modeling and response selection yield consistent gains in both belief tracking and long-horizon dialogue outcomes over strong LLM-based baselines.

2 Related Work

2.1 Personalized Dialogue Systems

Personalized dialogue systems aim to generate responses that remain coherent, adaptive, and user-specific across long-term interactions. Existing studies can be broadly categorized into profile-based personalization, memory-augmented dialogue systems, long-context personalization, and latent-variable dialogue modeling. Profile-, memory-, and long-context-based methods represent users with persona descriptions, preference profiles, structured attributes [57, 32, 56, 40, 23], short- or long-term memories [49, 13, 48, 50, 58], or extended interaction histories [33, 1, 15, 37, 30, 51, 31], and inject them through conditioning, retrieval, or long-context prompting to reuse past interactions, stated preferences, decisions, and personal facts. In parallel, latent-variable and state-aware dialogue models introduce hidden representations to capture implicit factors such as user intent, emotion, preference, dialogue state, or conversational goals [57, 53, 41, 29, 28]. These methods move beyond purely explicit user records by recognizing that observable utterances may only partially reveal the user's underlying condition. Existing methods mainly reuse user records or context, with limited support for action-conditioned latent state dynamics. PUMA explicitly models latent user state separately from memory, enabling active state-aware action selection beyond passive history reuse.

2.2 Free Energy Principle and Its Applications

The Free Energy Principle provides a principled account of adaptive behavior under uncertainty, viewing perception and action as a unified inference process over hidden states of the world [5, 10, 9]. Under this view, an agent updates its beliefs from observations and acts to make future observations more consistent with its preferences. Active inference operationalizes this process by selecting actions or policies according to expected free energy, which integrates goal-directed value with uncertainty reduction [7, 8]. This perspective has been extended to robotics, reinforcement learning, control, and AI agents, emphasizing belief updating and uncertainty-aware planning under partial observability [20, 12, 25, 4]. Recent work has further connected active inference with human-facing LLM and interactive systems, including reliable medical prompting [34], active preference learning and inference [21, 26, 18], and human–computer interaction modeling [24]. Existing FEP-inspired systems mainly apply active inference to external environments, task states, or relatively stable human-related variables, rarely treating the evolving user state itself as the modeling target. In contrast, PUMA applies FEP to personalized dialogue by making temporally evolving and action-sensitive user state the core object of active-inference modeling.

3 Preliminaries

The Free Energy Principle (FEP), originating from theoretical neuroscience and cognitive science, explains how biological agents adapt through perception, learning, and action when interacting with their external environment [6, 7]. This principle is commonly operationalized through two related objectives: variational free energy F_t (Section 3.1) for retrospective belief and internal world model updating over the present state, and expected free energy G_t (Section 3.2) for prospective action selection over future outcomes. We briefly introduce these two objectives below.

3.1 Variational Free Energy for Belief and world model Updating

FEP views an agent as interacting with a partially observed environment, where the hidden state s_t generates the observation o_t but cannot be directly accessed. The agent therefore maintains two internal quantities: a belief over the hidden state, representing its current uncertain understanding of the hidden state, and a world model of how the environment works. After receiving an observation, the agent updates its belief over the hidden state and refines its internal world model to better explain the observation. Variational free energy F_t provides a scalar objective for this process. In general, it evaluates how well the agent's approximate belief $q(s_t)$ and world model $p(o_t, s_t)$ explain observations under uncertainty:

$$F_t = \mathbb{E}_{q(s_t)} \left[\log q(s_t) - \log p(o_t, s_t) \right]. \tag{1}$$

Minimizing F_t supports belief updating and world model learning, enabling more plausible belief over user state and a more accurate account of the environment's underlying dynamics.

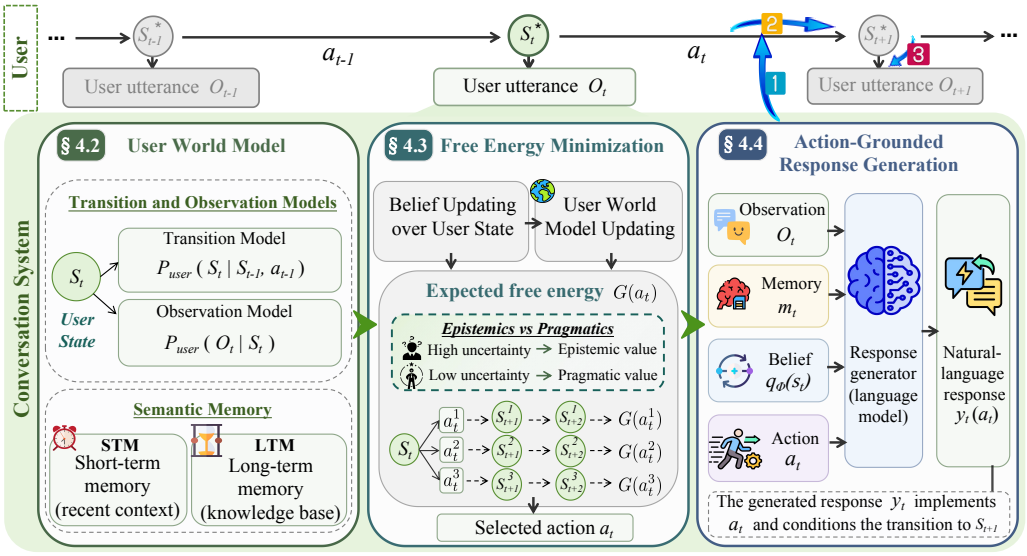


Figure 1: Overview of PUMA. Given a user utterance, the system updates its belief over the latent user state and maintains a user world model of its action-conditioned dynamics. It then uses this user world model to evaluate candidate actions with expected free energy, selecting actions that reduce uncertainty and guide goal-directed outcomes. Finally, the system produces a natural-language response, which gives rise to a user-state transition and the next user utterance, thereby starting the next interaction cycle.

3.2 Expected Free Energy for Action Selection

While variational free energy concerns updating beliefs and models after observations are received, action selection requires reasoning about outcomes that have not yet occurred. Expected free energy $G_t(\pi)$ extends the free-energy objective to this prospective setting by evaluating the anticipated consequences of candidate policies under the agent's belief and world model.

For each candidate policy π, the agent predicts possible future states and observations, and estimates whether the policy is expected to reduce uncertainty and lead to goal-directed outcomes. A policy with lower expected free energy is therefore expected to be both informative and goal-directed:

$$\pi^\star = \arg\min_{\pi} G_t(\pi). \tag{2}$$

Through this, expected free energy provides a general objective for action selection under uncertainty.

4 Method

Personalization is not merely about maintaining observable continuity in dialogue content; it also requires accounting for latent user states that influence user interpretation and response. We therefore propose **PUMA**, which models the user as a partially observable dynamical system, maintains a belief over the user's latent state, models its action-conditioned dynamics through a user world model, and selects actions that balance epistemic and pragmatic objectives. We first introduce our overall framework in Section 4.1, and then detail the user world model, belief updating, user world model updating, action selection, and response generation in Sections 4.2–4.4.

4.1 Overview: Framework of PUMA

In personalized dialogue, some important user-specific factors are implicit in language, making personalization a decision-making problem under partial observability. PUMA addresses this problem by maintaining a belief over the user's latent state, refining a user world model through interaction, and selecting actions according to their expected consequences. Figure 1 illustrates the overall framework, which includes three main parts: user world model, free energy minimization, and action-grounded response generation.

User World Model. Section 4.2 defines the user world model, which consists of the latent user state and the transition and observation models that capture its action-conditioned dynamics; a semantic memory outside the model supplies content-level evidence for personalization.

Free Energy Minimization. Given this model, Section 4.3 describes how the system updates its belief over the user's latent state, refines the user world model itself, and selects actions by balancing uncertainty reduction with goal-directed dialogue behavior.

Action-Grounded Response Generation. Given the selected action and updated belief over the user's latent state, Section 4.4 explains how the system generates a natural-language response grounded in the current observation, retrieved memory, and future-oriented dialogue action.

4.2 User World Model

At the core of PUMA is the user world model: a predictive model over user observations, latent user states, and their action-conditioned dynamics, which represents the user as a partially observable dynamical system. It consists of a latent user state s_t, a transition model of how s_t evolves under system actions, and an observation model of how s_t is expressed in language. The system's belief over s_t is inferred under this model (Section 4.3), and semantic memory, which stores content rather than dynamics, is kept outside it.

User State Personalized dialogue requires modeling not only content-level user information but also the user's evolving internal condition, which may shape utterances without being explicitly stated. Let s_t denote the latent user-state variable at turn t, with s_t^* its true value; this state represents the user's current decision-relevant condition. Since s_t^* is not directly accessible to the system, our system maintains a belief $q_\phi(s_t)$ over s_t, inferred from the dialogue history and the current observation. The user state s_t evolves over time and is action-relevant, enabling the system to interpret the current observation o_t beyond its explicit content and allowing system actions to shape its future trajectory.

Transition and Observation Models Formally, we factorize the core dynamics of the user world model as

$$p_\theta(o_t, s_t \mid s_{t-1}, a_{t-1}) = p_\theta(s_t \mid s_{t-1}, a_{t-1}) p_\theta(o_t \mid s_t). \tag{3}$$

This factorization separates two parts: 1) $p_\theta(s_t \mid s_{t-1}, a_{t-1})$, the transition model, captures action-conditioned state evolution; 2) $p_\theta(o_t \mid s_t)$, the observation model, captures how the latent user state generates the observed utterance.

The transition model $p_\theta(s_t \mid s_{t-1}, a_{t-1})$ describes how the latent user state evolves across turns under system actions. By modeling the effect of action a_{t-1} on the next state, it provides both the predictive component for future-state forecasting and the control target for action selection.

The observation model $p_\theta(o_t \mid s_t)$ draws how a user state is expressed as an observable utterance. It links the latent state to the actual response o_t, providing the inferential component for updating the system's belief over s_t.

Together with the latent user state, these two components constitute the user world model and support the action selection described in Section 4.3.

Semantic Memory The latent user state s_t captures the user's evolving, action-relevant condition, but does not store concrete semantic content. We therefore maintain a two-level semantic memory $M_t = \{M_t^{\mathrm{STM}}, M_t^{\mathrm{LTM}}\}$, where short-term memory M_t^{STM} stores session-local context and long-term memory M_t^{LTM} stores cross-session preferences, histories, goals, and behavioral regularities.

At each turn, the system retrieves relevant memory conditioned on the current belief $q_\phi(s_t)$, selected action a_t, and dialogue context $o_{\leq t}$:

$$m_t = \mathrm{Retrieve}(\{M_t^{\mathrm{STM}}, M_t^{\mathrm{LTM}}\} \mid q_\phi(s_t), a_t, o_{\leq t}). \tag{4}$$

The retrieved memory m_t grounds response generation with content-level personalization while keeping user-state tracking and the user world model separate from semantic storage.

4.3 Free Energy Minimization

Based on the user world model above, the system performs a Free Energy Minimization over turns. It first updates the belief over the latent user state from the new observation, then refines the user world model, and finally selects actions through the evaluation of future user-state trajectories.

Belief Updating over User State Because the true user state s_t^* is unobserved, the system maintains a variational belief $q_\phi(s_t)$ over the latent user-state variable s_t, inferred from the history and the current utterance o_t. Instantiating the general variational-free-energy objective in Eq. 1 with the user world model in Eq. 3, the turn-level belief update is defined as

$$q_\phi(s_t) = \arg\min_q \left[D_{\mathrm{KL}}\big(q(s_t) \,\|\, \tilde{p}_\theta(s_t)\big) - \mathbb{E}_{q(s_t)}\big[\log p_\theta(o_t \mid s_t)\big] \right], \tag{5}$$

where $\tilde{p}_\theta(s_t)$ denotes the predictive prior over the current user state induced by the previous belief and the previous system action. The derivation of Eq. 5, as well as the justification for introducing $q_\phi(s_t)$ under the user world model, is provided in Appendix B.1. The KL term preserves consistency with predicted state dynamics, while the likelihood term grounds the update in the observed utterance, enabling belief updating to balance dynamic plausibility with explanatory accuracy.

User World Model Updating After observing o_t and updating $q_\phi(s_t)$, the system refines the user world model using the discrepancy between predicted and observed user responses. The user world model refinement objective can be obtained by optimizing the observation generation ($p_\theta(o_t|s_t)$) and the action-conditioned state transition ($p_\theta(s_{t+1}|s_t, a_t)$). Specifically, we minimize

$$\mathcal{L}_{\mathrm{WM}} = \sum_{t=1}^{T} \mathbb{E}_{q_\phi(s_t)} \left[-\log p_\theta(o_t \mid s_t) \right] + \lambda \sum_{t=1}^{T-1} \mathbb{E}_{q_\phi(s_t)} \left[-\log p_\theta(s_{t+1} \mid s_t, a_t) \right], \tag{6}$$

The two terms measure observation-generation and action-conditioned transition errors, respectively, with λ balancing their contributions. This objective continuously refines how the user world model links user state, user utterance, and system action, enabling more accurate prediction of user dynamics. The derivation from the formulation in Eq. 5 is provided in Appendix B.2.

Action Selection through Expected Free Energy Given the current belief $q_\phi(s_t)$ and the learned user world model, action selection can be seen as model-based decision-making over user evolution: comparing candidate actions by their predicted consequences and choosing the one that best balances uncertainty reduction and goal-directed behavior. Following the expected-free-energy principle introduced in Section 3.2, we instantiate an approximation of $G(a)$ using the learned user world model for prospective action selection,

$$G(a) = \underbrace{\mathbb{E}_{q(o_{t+1}|a)} \left[H(q(s_{t+1} \mid o_{t+1}, a)) \right]}_{\text{epistemic term}} + \underbrace{\mathbb{E}_{q(s_{t+1}, o_{t+1}|a)} \left[-\log p_{\mathrm{pref}}(o_{t+1}) \right]}_{\text{pragmatic term}}. \tag{7}$$

The detailed derivation of this objective is provided in Appendix B.3. The system then selects the action that minimizes expected free energy: $a = \arg\min_{a \in \mathcal{A}} G(a)$. Here, $H(\cdot)$ denotes entropy, measuring the remaining uncertainty about the user's next state after action a and the predicted user response. *The epistemic term* favors actions that are expected to reduce uncertainty about the user. *The pragmatic term* favors actions whose predicted consequences are more aligned with the system's preferred outcomes, corresponding to pragmatic value. Thus, action selection is dynamically calibrated by the system's uncertainty about the user, providing an adaptive control mechanism that is unavailable to personalization methods without a user world model.

4.4 Action-Grounded Response Generation

The selected action a_t specifies the dialogue-level behavior, which is then realized as a natural-language response grounded in the belief and personalized content. The response is generated conditioned on the selected action, the belief over the user state, retrieved memory, and the current user utterance:

$$y_t \sim p_\psi(y_t \mid a_t, q_\phi(s_t), m_t, o_t). \tag{8}$$

Here, a_t provides explicit control, $q_\phi(s_t)$ gives the system's belief over the current user state, and m_t supplies content-level personalization for grounding the response. The generated response acts as an intervention on the latent user state and initiates the next interaction turn.

5 Experiments

In this section, we conduct experiments to answer the following research questions (RQs): **RQ1**: Can PUMA select actions that better uncover user needs and steer the dialogue toward desired future outcomes? **RQ2**: How do the three Free Energy Minimization components—belief updating over the current user state, user world model updating, and EFE-based action selection—contribute to the overall performance? **RQ3**: How well does the proposed user-state modeling mechanism support current-state estimation and next-state prediction across dialogue turns? **RQ4**: Can the proposed framework generalize its user-state estimation and transition prediction ability to a different dataset?

5.1 Experimental Setup

Dataset. We evaluate on CAMI (Counseling and Motivational Interviewing) [46], which contains 38 client profiles with complete MI counseling sessions. Each session is annotated with per-turn stage labels following the Transtheoretical Model [27]: *precontemplation*, *contemplation*, and *preparation*. Each profile includes a warmup session for persona construction and subsequent sessions for evaluation. For dynamic evaluation, we use the precontemplation-start profiles after removing duplicated profiles from the original set. For offline state-inference evaluation, we use the full profile set, yielding 427 evaluation turns after warmup.

Simulator-based dynamic evaluation. Since fixed offline transcripts cannot reveal how clients respond to different counselor policies, we conduct dynamic evaluation with *DynPatient*, a profile-grounded client simulator built from CAMI profiles and annotated sessions. Given the client profile, current stage, dialogue history, and counselor response, the simulator updates the client state and generates the next utterance. Details and validation are provided in Appendix D.

Compared Methods. We compare PUMA against representative counselor agent baselines. *DIIR* [43] uses information integration and reflection. *CoS* [35] is a chain-of-suggestion counselor. *CAMI-base*, *CAMI-text* and *CAMI-full* [46] are the basic, text-only and full CAMI counselors.

Evaluation Metrics. We evaluate the framework from two perspectives: dynamic counseling effectiveness and counselor-side response quality. For dynamic counseling effectiveness, we report *Stage Lift (Lift)*, *Prep Rate (Prep)*, *Trigger Coverage (TrigCov)*, and *Avg Turns (Turns)*. For counselor-side quality, we use the *MITI Global Scores* [46], including *Cultivating Change Talk*, *Softening Sustain Talk*, *Partnership*, and *Empathy*, and average them as *MITI-Avg*. Detailed metric definitions are provided in Appendix A.

Implementation Details. All experiments use *Qwen3-8B* [44] and *Llama-3.1-8B* [11] as backbone LLMs via local HuggingFace inference. For dynamic evaluation, all counselor methods interact with the same fixed client simulator, DynPatient. To make the system's representation of the user state directly evaluable in practice, we instantiate the user-state space with gold-labeled state annotations from existing datasets. All results use deterministic greedy decoding with fixed random seeds. In implementation, PUMA is realized through a set of fixed LLM-based inference modules that instantiate the framework's core components. The LLM judge for simulator validation and counselor-side quality evaluation uses Qwen3-32B in 4-bit quantization. The complete PUMA prompt templates are provided in Appendix G.1. The prompt templates for the long-prompt state-inference baseline in RQ3 are provided in Appendix G.4. Additional hyperparameter details are provided in Appendix C, and simulator details are provided in Appendix D.

5.2 Main Results (RQ1)

In this section, we evaluate whether PUMA can improve dynamic MI-coaching effectiveness. We focus on whether the model can select appropriate counseling actions, and guide the client toward goal-directed states. We report the main dynamic counseling results in Table 1, followed by counselor-side quality evaluation in Table 2. All results are obtained under the same CAMI evaluation protocol.

Dynamic counseling effectiveness. Table 1 shows that PUMA achieves the best dynamic counseling performance. Compared with DIIR, it improves Lift, Prep, and TrigCov while using fewer Turns.

This suggests that state-aware action selection enables more efficient progression toward the desired user state.

Counselor-side MI quality. Table 2 shows that PUMA receives the highest LLM-judge MITI scores among automated methods and remains comparable to the human-reference responses under this evaluation protocol. Overall, PUMA improves motivational outcomes while preserving a collaborative and empathetic counseling style.

Table 1: Dynamic evaluation on CAMI under Qwen3-8B and Llama-3.1-8B backbones. Higher values are better for Lift, Prep, and TrigCov, while lower values are better for Turns.

Counselor	Qwen3-8B				Llama-3.1-8B			
	Lift↑	Prep↑	TrigCov↑	Turns↓	Lift↑	Prep↑	TrigCov↑	Turns↓
CoS [35]	0.17	7.0%	5.0%	19.8	0.07	3.0%	3.0%	19.7
DIIR [43]	1.07	52.0%	34.0%	13.1	1.17	59.0%	40.0%	11.9
CAMI-base [46]	0.62	28.0%	22.0%	16.8	0.38	17.0%	14.2%	18.3
CAMI-text [46]	0.76	31.0%	27.0%	16.8	1.10	52.0%	34.9%	13.9
CAMI-full [46]	0.62	28.0%	23.0%	17.0	0.72	34.0%	24.5%	15.8
PUMA	**1.62**	**75.9%**	**62.4%**	**12.2**	**1.76**	**83.0%**	**63.2%**	**10.7**

Table 2: Counselor-side quality evaluation using MITI Global Scores on CAMI. We report four MI dimensions and their average score; higher scores indicate better counseling quality.

Method	Qwen3-8B					Llama-3.1-8B				
	CultCT↑	SoftST↑	Partner↑	Empathy↑	Avg↑	CultCT↑	SoftST↑	Partner↑	Empathy↑	Avg↑
CoS	1.31	1.41	1.41	1.76	1.47	1.24	1.28	1.24	1.69	1.36
DIIR	1.97	2.45	2.41	2.52	2.34	2.03	2.52	2.48	2.45	2.37
CAMI-base	1.59	2.00	2.00	2.03	1.91	1.55	2.00	1.93	1.90	1.84
CAMI-full	1.72	1.72	1.72	1.79	1.74	1.72	1.83	1.76	1.72	1.76
CAMI-text	1.83	1.97	1.86	1.83	1.87	2.07	2.21	2.14	2.14	2.14
Real (human)	3.38	4.10	4.00	3.93	3.85	3.38	4.10	4.00	3.93	3.85
PUMA	**3.59**	**4.59**	**4.66**	**4.62**	**4.37**	**3.62**	**4.45**	**4.55**	**4.45**	**4.27**

5.3 Ablation Study (RQ2)

Table 3 evaluates the three components of Free Energy Minimization (Section 4.3): belief updating over the current user state, user world model updating, and action selection through expected free energy. Removing the current-state belief estimate, the Transition and Observation Models, or EFE-based action selection each reduces Lift and Prep, while jointly removing the Transition and Observation Models and EFE-based action selection produces the largest degradation. Although *"PUMA w/o WM input"* achieves slightly higher TrigCov, its lower Lift and Prep indicate that this gain does not translate into better stage progression. These results support the complementary contributions of the three components to dynamic dialogue control.

5.4 In-depth Analysis (RQ3)

In this section, we further analyze why PUMA improves dynamic MI-coaching effectiveness. Different from the main experiments, this section focuses on whether the proposed framework provides a reliable user-state mechanism for action selection. We examine three aspects: current-state estimation, next-state prediction, and dynamic readiness progression.

State inference and temporal dynamics. Table 4 shows that PUMA achieves the best performance on both Curr-Acc and Next-Acc, demonstrating that it can accurately estimate the user's current motivational state while predicting how the state may evolve next. Figure 2a further shows stable next-state tracking across dialogue stages, with accuracy increasing from 0.613 at turns 0–2 to 0.757 after turn 10, suggesting that accumulated interaction evidence leads to more reliable state

Table 3: Ablation of the three Free Energy Minimization components on CAMI. Belief denotes estimation of the current user state, WM denotes the Transition and Observation Models, and EFE expected-free-energy action selection; ΔLift is computed relative to the full PUMA model.

Configuration	Belief	WM	EFE	Lift↑	Prep%↑	TrigCov↑	ΔLift
CAMI-base	✗	✗	✗	0.62	28.0%	22.0%	—
DIIR	partial	✗	✗	1.07	52.0%	34.0%	—
PUMA w/o WM+EFE	✓	✗	✗	1.35	55.2%	54.0%	−0.28
PUMA w/o Belief	✗	✓	✓	1.41	62.1%	56.6%	−0.21
PUMA w/o WM input	✓	✗†	✓	1.45	62.1%	**63.8%**	−0.17
PUMA w/o EFE	✓	✓	✗	1.41	62.1%	54.9%	−0.21
PUMA	✓	✓	✓	**1.62**	**75.9%**	62.4%	—

† The Transition and Observation Models are withheld from action selection, while EFE-based action selection remains enabled.

estimates. Meanwhile, Figure 2b shows that PUMA continues to improve client readiness throughout the interaction, whereas baselines tend to plateau after an initial increase, highlighting the benefit of state-aware action selection for long-horizon dialogue progression.

Case Study. We provide a qualitative case study illustrating how belief tracking supports dynamic action selection, with full interaction details reported in Appendix E.

Table 4: Current- and next-state accuracy.

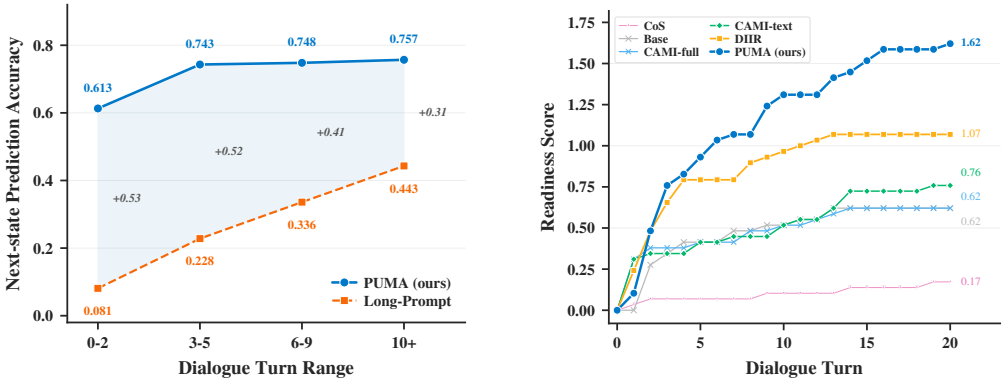
Method	Curr-Acc↑	Next-Acc↑
CAMI-full	0.504	–
Long-prompt	0.667	0.231
PUMA	**0.689**	**0.717**

Table 5: Generalization results on AnnoMI.

Method	Model	Curr-Acc↑	Next-Acc↑
Long-prompt	Qwen3-8B	0.463	0.364
Long-prompt	Qwen3-32B	0.602	0.430
PUMA	Qwen3-8B	**0.639**	**0.532**

5.5 Cross-Dataset Generalizability

We further test whether the proposed user world modeling framework generalizes beyond CAMI through a cross-dataset study on AnnoMI [42]. The detailed setup is provided in Appendix F. As shown in Table 5, PUMA with Qwen3-8B achieves higher Curr-Acc and Next-Acc than the long-prompt baseline with Qwen3-32B, despite using a smaller backbone. This result highlights the strength and scalability of the proposed user world modeling framework: even with a smaller backbone, PUMA achieves stronger current-state estimation and next-state prediction than a much larger long-prompt model. These results suggest that the framework can transfer across different user-state annotation schemes.



(a) Next-state prediction accuracy.

(b) Dynamic readiness progression.

Figure 2: Temporal analysis of state prediction and readiness progression.

6 Conclusion

We presented **PUMA**, an FEP-grounded framework for personalized dialogue. Instead of treating personalization as passive reuse of explicit user histories, our framework models the user as a partially observed dynamical system. This goes beyond static state estimation by explicitly modeling state transitions induced by dialogue actions. By maintaining a belief over the latent user state, refining a user world model, and selecting actions through expected free energy, PUMA enables the system to reason about both the current interpretation of the user and the future consequences of its responses.

Experiments in healthcare-oriented motivational interviewing settings show that PUMA improves dynamic counseling effectiveness, maintains strong counselor-side response quality, and provides reliable estimation. Ablation results support the contributions of the three Free Energy Minimization components to dynamic dialogue control, while cross-dataset results demonstrate the transferability of user-state estimation and prediction. These findings suggest that personalized dialogue systems can move beyond memory retrieval toward active, state-aware interaction control.

References

[1] Hongru Cai, Yongqi Li, Wenjie Wang, Fengbin Zhu, Xiaoyu Shen, Wenjie Li, and Tat-Seng Chua. Large language models empowered personalized web agents. In *Proceedings of the ACM on Web Conference 2025*, pages 198–215, 2025.

[2] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. A survey on dialogue systems: Recent advances and new frontiers. *Acm Sigkdd Explorations Newsletter*, 19(2):25–35, 2017.

[3] Itsugun Cho, Dongyang Wang, Ryota Takahashi, and Hiroaki Saito. A personalized dialogue generator with implicit user persona detection. *arXiv preprint arXiv:2204.07372*, 2022.

[4] Jingcheng Fang, Ying He, F Richard Yu, Jianqiang Li, and Victor C Leung. Large language models (llms) inference offloading and resource allocation in cloud-edge networks: An active inference approach. In *2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall)*, pages 1–5. IEEE, 2023.

[5] Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138, 2010.

[6] Karl Friston, James Kilner, and Lee Harrison. A free energy principle for the brain. *Journal of Physiology-Paris*, 100(1–3):70–87, 2006. doi: 10.1016/j.jphysparis.2006.10.001.

[7] Karl Friston, Francesco Rigoli, Dimitri Ognibene, Christoph Mathys, Thomas Fitzgerald, and Giovanni Pezzulo. Active inference and epistemic value. *Cognitive neuroscience*, 6(4):187–214, 2015.

[8] Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, Giovanni Pezzulo, et al. Active inference and learning. *Neuroscience & Biobehavioral Reviews*, 68:862–879, 2016.

[9] Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active inference: A process theory. *Neural Computation*, 29(1):1–49, 2017. doi: 10.1162/NECO_a_00912.

[10] Karl J Friston, Jean Daunizeau, James Kilner, and Stefan J Kiebel. Action and behavior: a free-energy formulation. *Biological cybernetics*, 102(3):227–260, 2010.

[11] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[12] Ying He, Jingcheng Fang, F Richard Yu, and Victor C Leung. Large language models (llms) inference offloading and resource allocation in cloud-edge computing: An active inference approach. *IEEE Transactions on Mobile Computing*, 23(12):11253–11264, 2024.

[13] Zhaopei Huang, Qifeng Dai, Guozheng Wu, Xiaopeng Wu, Xubin Li, Tiezheng Ge, Wenxuan Wang, and Qin Jin. Mem-pal: Towards memory-based personalized dialogue assistants for long-term user-agent interaction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 31229–31237, 2026.

[14] Takashi Kodama, Ribeka Tanaka, and Sadao Kurohashi. Dialogue management by estimating user's internal state using movie recommendation dialogue. *Journal of Natural Language Processing*, 28(1):104–135, 2021.

[15] Seongyun Lee, Sue Hyun Park, Seungone Kim, and Minjoon Seo. Aligning to thousands of preferences via system message generalization. *Advances in Neural Information Processing Systems*, 37:73783–73829, 2024.

[16] Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. Hello again! llm-powered personalized agent for long-term dialogue. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5259–5276, 2025.

[17] Dongshuo Liu, Zhijing Wu, Dandan Song, and He-Yan Huang. A persona-aware llm-enhanced framework for multi-session personalized dialogue generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 103–123, 2025.

[18] Xiaohao Liu, Xiaobo Xia, Weixiang Zhao, Manyi Zhang, Xianzhi Yu, Xiu Su, Shuo Yang, See-Kiong Ng, and Tat-Seng Chua. L-mtp: Leap multi-token prediction beyond adjacent context for large language models. *Advances in Neural Information Processing Systems*, 38: 102569–102600, 2026.

[19] Zhenyi Lu, Wei Wei, Xiaoye Qu, XianLing Mao, Dangyang Chen, and Jixiong Chen. Miracle: Towards personalized dialogue generation with latent-space multiple personal attribute control. *arXiv preprint arXiv:2310.18342*, 2023.

[20] Siyuan Ma, Bo Gao, Xiaojun Jia, Simeng Qin, Tianlin Li, Ke Ma, Xiaoshuang Jia, Wenqi Ren, and Yang Liu. Odar: Principled adaptive routing for llm reasoning via active inference. *arXiv preprint arXiv:2602.23681*, 2026.

[21] Saaduddin Mahmud, Mason Nakamura, and Shlomo Zilberstein. Maple: A framework for active preference learning guided by large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 27518–27528, 2025.

[22] Shuyi Miao, Yaojin Ma, Chenhang Cui, Xiaohao Liu, Dang Jisheng, Shengda Zhuo, Fei Shen, and Tat-Seng Chua. Emorss: Mitigating emotion-induced over-refusal in large language models. *arXiv preprint arXiv:2610.04998*, 2026.

[23] Shuyi Miao, Wangjie Qiu, Shengda Zhuo, Fei Shen, Dan Lin, Xingtong Yu, Chua Tat-Seng, and Zhiming Zheng. Unidetect: Llm-driven universal fraud detection across heterogeneous blockchains. *arXiv preprint arXiv:2604.12329*, 2026.

[24] Roderick Murray-Smith, John H Williamson, and Sebastian Stein. Active inference and human–computer interaction. *ACM Transactions on Computer-Human Interaction*, 32(6):1–45, 2025.

[25] Giovanni Pezzulo, Thomas Parr, Paul Cisek, Andy Clark, and Karl Friston. Generating meaning: active inference and the scope and limits of passive ai. *Trends in Cognitive Sciences*, 28(2): 97–112, 2024.

[26] Top Piriyakulkij, Volodymyr Kuleshov, and Kevin Ellis. Active preference inference using language models and probabilistic reasoning. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2025.

[27] James O Prochaska, Susan Butterworth, Colleen A Redding, Verna Burden, Nancy Perrin, Michael Leo, Marna Flaherty-Robb, and Janice M Prochaska. Initial efficacy of mi, ttm tailoring and hri's with multiple behaviors for employee health promotion. *Preventive medicine*, 46(3):226–231, 2008.

[28] Yilun Qiu, Tianhao Shi, Xiaoyan Zhao, Fengbin Zhu, Yang Zhang, and Fuli Feng. Latent inter-user difference modeling for llm personalization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10610–10628, 2025.

[29] Yilun Qiu, Xiaoyan Zhao, Yang Zhang, Yimeng Bai, Wenjie Wang, Hong Cheng, Fuli Feng, and Tat-Seng Chua. Measuring what makes you unique: Difference-aware user modeling for enhancing llm personalization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 21258–21277, 2025.

[30] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D Manning. Raptor: Recursive abstractive processing for tree-organized retrieval. In *The Twelfth International Conference on Learning Representations*, 2024.

[31] Pengyang Shao, Le Wu, Kun Zhang, Defu Lian, Richang Hong, Yong Li, and Meng Wang. Average user-side counterfactual fairness for collaborative filtering. *ACM Transactions on Information Systems*, 42(5):1–26, 2024.

[32] Xiaoteng Shen, Rui Zhang, Xiaoyan Zhao, Jieming Zhu, and Xi Xiao. Pmg: Personalized multimodal response generation with large language models. In *The Web Conference 2024*.

[33] Yunxiao Shi, Haoning Shang, Xing Zi, Wujiang Xu, Yue Feng, and Min Xu. Answering narrative-driven recommendation queries via a retrieve–rank paradigm and the ocg-agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13192–13213, 2025.

[34] Roma Shusterman, Allison C Waters, Shannon O'Neill, Marshall Bangs, Phan Luu, and Don M Tucker. An active inference strategy for prompting reliable responses from large language models in medical practice. *npj Digital Medicine*, 8(1):119, 2025.

[35] Xin Sun, Xiao Tang, Abdallah El Ali, Zhuying Li, Xiaoyu Shen, Pengjie Ren, Jan de Wit, Jiahuan Pei, and Jos A Bosch. Chain-of-strategy planning with llms: Aligning the generation of psychotherapy dialogue with strategy in motivational interviewing. *arXiv preprint arXiv:2408.06527*, 2024.

[36] Yihong Tang, Bo Wang, Miao Fang, Dongming Zhao, Kun Huang, Ruifang He, and Yuexian Hou. Enhancing personalized dialogue generation with contrastive latent variables. *arXiv preprint arXiv:2305.11482*, 2023.

[37] Bohao Wang, Feng Liu, Jiawei Chen, Xingyu Lou, Changwang Zhang, Jun Wang, Yuegang Sun, Yan Feng, Chun Chen, and Can Wang. Msl: Not all tokens are what you need for tuning llm as a recommender. In *Proceedings of the 48th international ACM SIGIR conference on research and development in information retrieval*, pages 1912–1922, 2025.

[38] Chengbing Wang, Yang Zhang, Wenjie Wang, Xiaoyan Zhao, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Think-while-generating: On-the-fly reasoning for personalized long-form generation. *arXiv preprint arXiv:2512.06690*, 2025.

[39] Zheng Wang, Zhongyang Li, Zeren Jiang, Dandan Tu, and Wei Shi. Crafting personalized agents through retrieval-augmented generation on editable memory graphs. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 4891–4906, 2024.

[40] Jiaxing Wu, Lin Ning, Luyang Liu, Harrison Lee, Neo Wu, Chao Wang, Sushant Prakash, Shawn O'Banion, Bradley Green, and Jun Xie. Rlpf: Reinforcement learning from prediction feedback for user summarization with llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25488–25496, 2025.

[41] Shirley Wu, Evelyn Choi, Arpandeep Khatua, Zhanghan Wang, Joy He-Yueya, Tharindu Cyril Weerasooriya, Wei Wei, Diyi Yang, Jure Leskovec, and James Zou. Humanlm: Simulating users with state alignment beats response imitation. *arXiv preprint arXiv:2603.03303*, 2026.

[42] Zixiu Wu, Simone Balloccu, Vivek Kumar, Rim Helaoui, Ehud Reiter, Diego Reforgiato Recupero, and Daniele Riboni. Anno-mi: A dataset of expert-annotated counselling dialogues. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6177–6181. IEEE, 2022.

[43] Zhouhang Xie, Bodhisattwa Prasad Majumder, Mengjie Zhao, Yoshinori Maeda, Keiichi Yamada, Hiromi Wakaki, and Julian McAuley. Few-shot dialogue strategy learning for motivational interviewing via inductive reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13207–13219, 2024.

[44] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[45] Xinyu Yang, Junlin Han, Rishi Bommasani, Jinqi Luo, Wenjie Qu, Wangchunshu Zhou, Adel Bibi, Xiyao Wang, Jaehong Yoon, Elias Stengel-Eskin, et al. Reliable and responsible foundation models. *Transactions on Machine Learning Research*, 2025.

[46] Yizhe Yang, Palakorn Achananuparp, He-Yan Huang, Jing Jiang, Phey Ling Kit, Nicholas Gabriel Lim, Cameron Tan Shi Ern, and Ee-Peng Lim. Cami: A counselor agent supporting motivational interviewing through state inference and topic exploration. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 21037–21081, 2025.

[47] Yizhe Yang, Palakorn Achananuparp, He-Yan Huang, Jing Jiang, Nicholas Gabriel Lim, Cameron Tan Shi Ern, Phey Ling Kit, Jenny Giam Xiuhui, John Pinto, and Ee-peng Lim. Consistent client simulation for motivational interviewing-based counseling. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20959–20998, 2025.

[48] Shuo Yu, Mingyue Cheng, Daoyu Wang, Qi Liu, Zirui Liu, Ze Guo, and Xiaoyu Tao. Memweaver: A hierarchical memory from textual interactive behaviors for personalized generation. In *Proceedings of the ACM Web Conference 2026*, pages 6920–6931, 2026.

[49] Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. Memskill: Learning and evolving memory skills for self-evolving agents. *arXiv preprint arXiv:2602.02474*, 2026.

[50] Xinliang Frederick Zhang, Nick Beauchamp, and Lu Wang. Prime: Large language model personalization with cognitive dual-memory and personalized thought process. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 33695–33724, 2025.

[51] Yang Zhang, Wenxin Xu, Xiaoyan Zhao, Wenjie Wang, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Reinforced latent reasoning for llm-based recommendation. *ICLR*, 2026.

[52] Zeyu Zhang, Rui Li, Xiaoyan Zhao, Yang Zhang, Wenjie Wang, Xu Chen, and Tat-Seng Chua. Nextmem: Towards latent factual memory for llm-based agents. *arXiv preprint arXiv:2603.15634*, 2026.

[53] Zeyu Zhang, Yang Zhang, Haoran Tan, Rui Li, and Xu Chen. Explicit vs implicit memory: Exploring multi-hop complex reasoning over personalized information. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pages 1964–1974, 2026.

[54] Xiaoyan Zhao, Yang Deng, Wenjie Wang, Hong Cheng, Rui Zhang, See-Kiong Ng, Tat-Seng Chua, et al. Exploring the impact of personality traits on conversational recommender systems: A simulation with large language models. *arXiv preprint arXiv:2504.12313*, 2025.

[55] Xiaoyan Zhao, Ming Yan, Yilun Qiu, Haoting Ni, Yang Zhang, Fuli Feng, Hong Cheng, and Tat-Seng Chua. Steerx: Disentangled steering for llm personalization. *arXiv preprint arXiv:2510.22256*, 2025.

[56] Xiaoyan Zhao, Ming Yan, Yang Zhang, Yang Deng, Jian Wang, Fengbin Zhu, Yilun Qiu, Hong Cheng, and Tat-Seng Chua. Reinforced strategy optimization for conversational recommender systems via network-of-experts. *arXiv e-prints*, pages arXiv–2509, 2025.

[57] Xiaoyan Zhao, Juntao You, Yang Zhang, Wenjie Wang, Hong Cheng, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. Nextquill: Causal preference modeling for enhancing llm personalization. *ICLR*, 2026.

[58] Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 19724–19731, 2024.

A Evaluation Metrics

We evaluate the framework from three perspectives: static state inference, dynamic counseling effectiveness, and counselor-side response quality.

Dynamic counseling effectiveness. **Stage Lift** measures the average ordinal improvement from the initial to the final motivational stage. **Prep Rate** is the percentage of profiles that reach the preparation stage. **Trigger Coverage** measures the fraction of profile-specific motivational triggers addressed by the counselor via semantic matching. **Avg Turns** reports the average number of turns used to complete the session, where lower values indicate higher efficiency when stage progression is strong.

Counselor-side response quality. We use four **MITI Global Scores** [46]. **Cultivating Change Talk** measures the extent to which the counselor encourages client language favoring change. **Softening Sustain Talk** measures the extent to which the counselor avoids amplifying resistance. **Partnership** measures whether the counselor conveys that expertise and choice reside within the client. **Empathy** measures whether the counselor understands and reflects the client's perspective. Each score is rated on a 1–5 Likert scale. We report the average of the four scores as **MITI-Avg**.

Static state inference. **Curr-Acc** measures the proportion of turns where the predicted current motivational state matches the gold label. **Next-Acc** measures the accuracy of predicting the client's motivational state at the following turn.

B Technical Appendices

B.1 Derivation of the Posterior Inference Objective

We derive the posterior inference objective from the standard free-energy construction. At turn t, the user world model factorizes the joint distribution of the current observation and latent user state as

$$p_\theta(o_t, s_t \mid s_{t-1}, a_{t-1}) = p_\theta(o_t \mid s_t) p_\theta(s_t \mid s_{t-1}, a_{t-1}). \tag{9}$$

For notational simplicity, we omit the conditioning on previous states and actions in the following derivation, and write the current-turn joint distribution as $p_\theta(o_t, s_t)$.

Following the Free Energy Principle, the key goal can be formulated as reducing the surprise of observations under the agent's generative model. For the current user observation o_t, this surprise is given by

$$-\log p_\theta(o_t). \tag{10}$$

However, evaluating this quantity requires computing the marginal likelihood, or model evidence, of the observation:

$$p_\theta(o_t) = \int p_\theta(o_t, s_t)\, ds_t. \tag{11}$$

This term marginalizes over all possible latent user states that could have generated the current observation. Since the latent state space is generally large or continuous, directly computing $p_\theta(o_t)$, and hence the surprise $-\log p_\theta(o_t)$, is generally intractable.

To obtain a tractable objective, we introduce an auxiliary distribution $q(s_t)$ over the latent state. This introduction does not change the marginal likelihood, since

$$p_\theta(o_t) = \int p_\theta(o_t, s_t)\, ds_t \tag{12}$$

$$= \int q(s_t) \frac{p_\theta(o_t, s_t)}{q(s_t)}\, ds_t \tag{13}$$

$$= \mathbb{E}_{q(s_t)} \left[\frac{p_\theta(o_t, s_t)}{q(s_t)} \right]. \tag{14}$$

Substituting Eq. 14 into the surprise gives

$$-\log p_\theta(o_t) = -\log \mathbb{E}_{q(s_t)} \left[\frac{p_\theta(o_t, s_t)}{q(s_t)} \right]. \tag{15}$$

Since $-\log(\cdot)$ is convex, Jensen's inequality gives

$$-\log p_\theta(o_t) \leq \mathbb{E}_{q(s_t)}\left[-\log \frac{p_\theta(o_t, s_t)}{q(s_t)}\right] \tag{16}$$

$$= \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(o_t, s_t)\right]. \tag{17}$$

The right-hand side is the variational free energy:

$$\mathcal{F}_t(q) = \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(o_t, s_t)\right]. \tag{18}$$

Thus, variational free energy provides an upper bound on the surprise:

$$-\log p_\theta(o_t) \leq \mathcal{F}_t(q). \tag{19}$$

Following the Free Energy Principle, the agent avoids surprising observations by minimizing this upper bound. Since the true surprise $-\log p_\theta(o_t)$ is difficult to compute directly, the agent instead optimizes the tractable objective $\mathcal{F}_t(q)$ with respect to its belief $q(s_t)$:

$$q^\star(s_t) = \arg\min_q \mathcal{F}_t(q). \tag{20}$$

Using the current-turn factorization

$$p_\theta(o_t, s_t) = p_\theta(o_t \mid s_t) p_\theta(s_t), \tag{21}$$

we can rewrite the variational free energy as

$$\mathcal{F}_t(q) = \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(s_t) - \log p_\theta(o_t \mid s_t)\right] \tag{22}$$

$$= D_{\mathrm{KL}}\left(q(s_t) \,\|\, p_\theta(s_t)\right) - \mathbb{E}_{q(s_t)}\left[\log p_\theta(o_t \mid s_t)\right]. \tag{23}$$

This decomposition gives the posterior update objective used in our framework. The first term keeps the updated belief $q(s_t)$ close to the prior $p_\theta(s_t)$, which encodes the expected user state before observing o_t. The second term encourages $q(s_t)$ to place probability mass on latent states that explain the current observation through the likelihood $p_\theta(o_t \mid s_t)$. Therefore, minimizing $\mathcal{F}_t(q)$ updates the system's belief by balancing prior consistency and observation fit.

To show that this optimization corresponds to posterior inference, we can also rewrite the joint distribution as

$$p_\theta(o_t, s_t) = p_\theta(s_t \mid o_t) p_\theta(o_t). \tag{24}$$

Substituting this into Eq. 18 yields

$$\mathcal{F}_t(q) = \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(s_t \mid o_t) - \log p_\theta(o_t)\right] \tag{25}$$

$$= D_{\mathrm{KL}}\left(q(s_t) \,\|\, p_\theta(s_t \mid o_t)\right) - \log p_\theta(o_t). \tag{26}$$

The second term $-\log p_\theta(o_t)$ does not depend on $q(s_t)$. Therefore, minimizing $\mathcal{F}_t(q)$ with respect to $q(s_t)$ is equivalent to minimizing

$$D_{\mathrm{KL}}\left(q(s_t) \,\|\, p_\theta(s_t \mid o_t)\right). \tag{27}$$

Thus, $q(s_t)$ becomes an approximate posterior after optimization, rather than being assumed to equal the posterior from the beginning.

B.2 Derivation of the World Model Learning Objective

We derive the world model learning objective from the per-turn posterior inference objective established in Appendix B.1. Starting from the variational free energy at turn t,

$$\mathcal{F}_t(q, \theta) = D_{\mathrm{KL}}\left(q(s_t) \,\|\, \tilde{p}_\theta(s_t)\right) - \mathbb{E}_{q(s_t)}\left[\log p_\theta(o_t \mid s_t)\right], \tag{28}$$

we note that this objective depends on two sets of parameters: the variational parameters ϕ of the belief $q_\phi(s_t)$, and the generative parameters θ of the world model. Posterior inference (Eq. 5) optimizes $\mathcal{F}_t$ with respect to ϕ at a single turn. World model learning instead optimizes the same free energy with respect to θ, accumulated over the dialogue.

Aggregating Eq. 28 along the trajectory yields

$$\mathcal{F}(\theta) = \sum_{t=1}^{T} \mathcal{F}_t(q_\phi, \theta). \tag{29}$$

We now identify the θ-dependent components in Eq. 29.

Observation term. The likelihood term in Eq. 28 contributes, at each turn,

$$-\mathbb{E}_{q_\phi(s_t)}\left[\log p_\theta(o_t \mid s_t)\right], \tag{30}$$

which trains the observation model to assign high likelihood to the actual user utterance o_t under the inferred latent state.

Transition term. The KL term in Eq. 28 depends on θ only through the predictive prior

$$\tilde{p}_\theta(s_t) = \int p_\theta(s_t \mid s_{t-1}, a_{t-1})\, q_\phi(s_{t-1})\, ds_{t-1}, \tag{31}$$

which is determined by the action-conditioned transition model $p_\theta(s_t \mid s_{t-1}, a_{t-1})$. Reindexing this contribution from t to $t+1$ so that it expresses how state s_{t+1} is predicted from s_t and a_t, the θ-dependent part of the prior takes the form

$$-\mathbb{E}_{q_\phi(s_t)}\left[\log p_\theta(s_{t+1} \mid s_t, a_t)\right], \quad t = 1, \ldots, T-1, \tag{32}$$

which trains the transition model to predict the next inferred user state from the current state and system action.

Combined objective. Summing Eqs. 30 and 32 over the trajectory and introducing a coefficient λ to balance the relative scales of the observation and transition losses gives

$$\mathcal{L}_{\mathrm{WM}} = \sum_{t=1}^{T} \mathbb{E}_{q_\phi(s_t)}\left[-\log p_\theta(o_t \mid s_t)\right] + \lambda \sum_{t=1}^{T-1} \mathbb{E}_{q_\phi(s_t)}\left[-\log p_\theta(s_{t+1} \mid s_t, a_t)\right], \tag{33}$$

which is exactly Eq. 6. Thus, posterior inference and world model learning optimize the same per-turn variational free energy with respect to ϕ and θ respectively, with the latter aggregated along the dialogue trajectory.

B.3 Derivation of the Expected Free Energy Objective

The posterior inference objective in Appendix B.1 evaluates the variational free energy at the current turn, where the observation o_t is already available. For action selection, however, the agent must reason about a future turn $t+1$, at which the next observation o_{t+1} has not yet been received. We extend the free energy into an *expected free energy* that scores each candidate action a, starting from the equivalent form of the variational free energy obtained in Eq. 26:

$$\mathcal{F}_t(q) = D_{\mathrm{KL}}\left(q(s_t) \,\|\, p_\theta(s_t \mid o_t)\right) - \log p_\theta(o_t). \tag{34}$$

Since $-\log p_\theta(o_t)$ does not depend on s_t, it can be absorbed into the expectation under $q(s_t)$, yielding the joint-surprise form

$$\mathcal{F}_t(q) = \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(s_t \mid o_t) - \log p_\theta(o_t)\right] \tag{35}$$

$$= \mathbb{E}_{q(s_t)}\left[\log q(s_t) - \log p_\theta(s_t, o_t)\right]. \tag{36}$$

To plan an action a at turn t, we evaluate the analogous quantity at turn $t+1$ under three modifications. *(i)* The future observation o_{t+1} is unknown, so we take an additional expectation over the predicted observation distribution implied by the world model,

$$q(o_{t+1} \mid a) = \int q(s_{t+1} \mid a)\, p_\theta(o_{t+1} \mid s_{t+1})\, ds_{t+1}. \tag{37}$$

(ii) The current belief $q(s_t)$ is replaced by the prior-predictive belief $q(s_{t+1} \mid a)$, obtained by propagating $q(s_t)$ one step forward through the user transition model conditional on a. *(iii)* The model evidence $p_\theta(o_{t+1})$, which is a neutral marginal in the inference setting, is replaced by a preference distribution $p_{\mathrm{pref}}(o_{t+1})$ encoding the desired future outcomes. Equivalently, the joint distribution $p_\theta(s_{t+1}, o_{t+1} \mid a)$ is replaced by the preference-biased joint

$$\tilde{p}(s_{t+1}, o_{t+1} \mid a) = p_\theta(s_{t+1} \mid o_{t+1}, a)\, p_{\mathrm{pref}}(o_{t+1}). \tag{38}$$

This substitution is the central modeling step of active inference: it turns the inference objective into a planning objective by aligning low surprise with desired outcomes rather than with predictive accuracy alone.

Applying these three modifications to Eq. 36, we define the expected free energy under action a as

$$G(a) = \mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[\log q(s_{t+1} \mid a) - \log \tilde{p}(s_{t+1}, o_{t+1} \mid a) \right] \tag{39}$$

$$= \mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[\log q(s_{t+1} \mid a) - \log p_\theta(s_{t+1} \mid o_{t+1}, a) - \log p_{\text{pref}}(o_{t+1}) \right]. \tag{40}$$

The term $p_\theta(s_{t+1} \mid o_{t+1}, a)$ is the true posterior over the future user state, which is intractable for the same reason $p_\theta(s_t \mid o_t)$ is intractable in the current-turn case. Following the same variational principle, we approximate it by the variational posterior $q(s_{t+1} \mid o_{t+1}, a)$:

$$\begin{aligned} G(a) \approx\ & \mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[\log q(s_{t+1} \mid a) - \log q(s_{t+1} \mid o_{t+1}, a) \right] \\ & + \mathbb{E}_{q(o_{t+1} \mid a)} \left[- \log p_{\text{pref}}(o_{t+1}) \right]. \end{aligned} \tag{41}$$

The first term in Eq. 41 admits an information-theoretic interpretation. Factoring the joint expectation as $q(s_{t+1}, o_{t+1} \mid a) = q(o_{t+1} \mid a)\, q(s_{t+1} \mid o_{t+1}, a)$,

$$\begin{aligned} & \mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[\log q(s_{t+1} \mid a) - \log q(s_{t+1} \mid o_{t+1}, a) \right] \\ & = -\mathbb{E}_{q(o_{t+1} \mid a)} \left[D_{\text{KL}} \left(q(s_{t+1} \mid o_{t+1}, a) \,\|\, q(s_{t+1} \mid a) \right) \right] && (42) \\ & = -I(s_{t+1}; o_{t+1} \mid a), && (43) \end{aligned}$$

where $I(s_{t+1}; o_{t+1} \mid a)$ is the mutual information between the future user state and the future observation conditional on action a. This quantity, the *epistemic value* of the action, measures how much the agent expects to learn about s_{t+1} by observing the user response o_{t+1} after taking action a. Equivalently, the mutual information decomposes into a difference of entropies:

$$I(s_{t+1}; o_{t+1} \mid a) = H(q(s_{t+1} \mid a)) - \mathbb{E}_{q(o_{t+1} \mid a)} \left[H(q(s_{t+1} \mid o_{t+1}, a)) \right]. \tag{44}$$

Substituting Eq. 44 back into Eq. 41 gives

$$\begin{aligned} G(a) \approx\ & \mathbb{E}_{q(o_{t+1} \mid a)} \left[H(q(s_{t+1} \mid o_{t+1}, a)) \right] - H(q(s_{t+1} \mid a)) \\ & + \mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[- \log p_{\text{pref}}(o_{t+1}) \right]. \end{aligned} \tag{45}$$

Action selection only requires $\arg\min_{a \in \mathcal{A}} G(a)$, so any term that is approximately invariant across candidate actions does not affect the decision. The prior-predictive entropy $H(q(s_{t+1} \mid a))$ is dominated by the current belief $q(s_t)$ propagated through the transition model and varies negligibly across candidate actions compared with the data-driven posterior entropy and the preference-weighted log-likelihood. Treating it as a constant absorbs it into the $\arg\min$ and yields the action-selection objective used in Section 4.3:

$$G(a) = \underbrace{\mathbb{E}_{q(o_{t+1} \mid a)} \left[H(q(s_{t+1} \mid o_{t+1}, a)) \right]}_{\text{exploration term}} + \underbrace{\mathbb{E}_{q(s_{t+1}, o_{t+1} \mid a)} \left[- \log p_{\text{pref}}(o_{t+1}) \right]}_{\text{exploitation term}}, \tag{46}$$

with the selected action $a_t = \arg\min_{a \in \mathcal{A}} G(a)$.

The exploration term is the expected entropy of the agent's belief about the next user state *after* taking action a and observing the resulting response, so minimizing it favors actions whose predicted observations are most informative about the user. The exploitation term is the expected negative log-preference of the predicted observation, so minimizing it favors actions whose consequences best match the desired future outcomes. Together, these two terms realize the active-inference principle that an agent should both reduce uncertainty about the user and steer interactions toward preferred states.

C Hyperparameter Settings

This appendix summarizes the hyperparameter settings used in our experiments. Unless otherwise specified, all experiments use the default values reported in Table 6. The core PUMA hyperparameters are shared by both the static and dynamic evaluation tracks, although they are passed through different configuration interfaces: the static track uses `config.py`, while the dynamic track uses script-level `argparse` arguments or internal default dictionaries. In large-scale runs, Slurm scripts may override these default values.

Table 6: General hyperparameters used in PUMA.

Hyperparameter	Default Value	Description
relevance_memory _number	1	Number of retrieved relevant memories.
context_memory _number	30	Maximum number of contextual memories.
dist_thres	1.5	Distance threshold for relevance-based retrieval.
persona_history _max	6	Maximum history turns for long-term persona modeling.
persona_short_term _window_turns	4	Short-term window size for q-state inference.
persona_long_term _summary_turns	12	Number of turns used for long-term p-state summary update.
fep_lambda_e / fep_lambda_p	0.4 / 0.6	Epistemic and pragmatic weights in EFE-based action selection.
fep_soft_fusion _beta	0.35	Weight of the planner prior in soft fusion.
max_output _length	1024	Maximum number of generated LLM tokens.
cami_warmup_ratio / cami_min_eval_turns	0.5 / 3	Warm-up and evaluation split for CAMI data.
max_turns	12	Maximum number of counselor turns in interactive_cami_text_eval.

Table 7: Hyperparameters for eval_sim.py.

Hyperparameter	Default Value	Description
max_turns	20	Maximum number of dialogue turns.
seed	42	Random seed.
test_num	5	Number of evaluated profiles by default.
fep_lambda_e / fep_lambda_p	0.4 / 0.6	Epistemic and pragmatic weights in EFE-based action selection.
disable_planner	False	Whether to disable the transition planner.
observation_focused _state	True	Whether to use the observation-focused state head.
fep_efe_action	True	Whether to use EFE-based action selection.
cami_preparation _calibration	True	Whether to apply preparation-stage calibration.
doctor_modes	fep, cami_text, base_counselor, diir, cos, cami_full	Compared counselor agents.

Shared PUMA Parameters. The two evaluation tracks share the same core PUMA parameters. Specifically, fep_lambda_e=0.4 and fep_lambda_p=0.6 control the trade-off between epistemic value and pragmatic value in expected-free-energy action selection. The soft-fusion coefficient fep_soft_fusion_beta=0.35 controls the contribution of the planner prior when combining model-based predictions with observation-driven state inference. These parameters are kept fixed across the main experiments unless explicitly changed in ablation or sensitivity analysis.

D Client Simulator Details and Validation

This appendix describes the design and validation of **DynPatient** , the client simulator used for simulator-based dynamic evaluation. The simulator provides a controlled environment for comparing counselor policies under the same client profiles and initial conditions. It is used only for dynamic

Table 8: Hyperparameters for `eval_patient_simulator.py`.

Hyperparameter	Default Value	Description
`max_turns`	20	Maximum number of dialogue turns.
`seed`	42	Random seed.
`judge_model`	`Qwen3-32B`	LLM judge used for simulator evaluation.
`load_in_4bit`	`False`	Whether to enable 4-bit quantized loading.

Table 9: Trigger construction in DynPatient.

Source	Length filter	Discovery bonus
Beliefs	len > 10	0.2
Motivation	len > 20	0.4
Acceptable Plans	len > 10	0.5

evaluation, while static state inference and next-state prediction are evaluated directly against gold annotated turns.

D.1 Simulator Design

Design goals. DynPatient is a profile-grounded and LLM-driven client simulator. Given a counselor utterance, the simulator first determines whether the response addresses profile-specific motivational triggers, then updates an internal readiness variable, performs stage transition, selects a client action, and finally generates a natural-language client response. The design follows three principles. First, the simulator should implement a learnable state machine, where client-state transitions are driven by counselor behavior rather than random sampling. Second, it should be profile-faithful, with each simulated client grounded in the corresponding CAMI profile, including beliefs, motivations, acceptable plans, personas, and behavior descriptions. Third, it should be reproducible, using deterministic greedy decoding and fixed random seeds.

State space. The simulator follows the Transtheoretical Model stage space used in CAMI:

$$\mathcal{S} = \{\textit{precontemplation}, \textit{contemplation}, \textit{preparation}\}. \tag{47}$$

Each episode starts from the initial stage given by the corresponding CAMI profile. The dynamic evaluation measures whether a counselor can move the simulated client toward *preparation* within a fixed turn budget.

Profile-grounded triggers. For each client profile, we construct a set of motivational triggers from the original profile sentences. We use sentences from *beliefs*, *motivations*, and *acceptable plans* as profile-specific anchors. Each trigger is represented as a tuple (c_i, z_i, b_i), where c_i denotes the source category, z_i is the trigger sentence, and b_i is its discovery bonus. Personas are used as background context for response generation, but are not included in the trigger set.

Semantic trigger matching. Given the counselor utterance u_t, the simulator computes semantic similarity between u_t and each profile trigger. We encode both trigger sentences and counselor utterances with a shared SentenceTransformer encoder and use cosine similarity for matching:

$$\mathcal{M}_t = \{i : \cos(\mathbf{e}_{z_i}, \mathbf{e}_{u_t}) \geq \tau\}, \qquad \tau = 0.45. \tag{48}$$

Here, $\mathcal{M}_t$ denotes the set of triggers matched at turn t. The threshold is set to avoid overly loose matches, so counselor responses must semantically address profile-specific content to affect the client's readiness.

Readiness dynamics. The simulator maintains a continuous readiness variable $r_t \in \mathbb{R}$, initialized as $r_0 = 0$. At each turn, readiness is updated as:

$$r_{t+1} = r_t + \underbrace{\mathbb{E}[\Delta r \mid s_t, a_t] \cdot g_t}_{\text{action-driven contribution}} + \underbrace{\sum_{i \in \mathcal{N}_t} b_i}_{\text{discovery bonus}}, \tag{49}$$

where s_t is the current client stage, a_t is the selected client action, g_t is a content gate, and $\mathcal{N}_t$ is the set of newly discovered triggers at turn t.

The expected readiness increment is estimated from CAMI annotations:

$$\mathbb{E}[\Delta r \mid s, a] = \sum_{tt \in \{c,n,su\}} P(tt \mid s, a) w_{tt}, \tag{50}$$

where tt denotes client talk type, including change talk c, neutral talk n, and sustain talk su. We use $w_c = 1.0$, $w_n = 0.3$, and $w_{su} = -1.0$. Only counselor-client turn pairs with sufficient support are used to estimate $P(tt \mid s, a)$.

Content gate. The content gate prevents generic counselor responses from increasing readiness without addressing profile-specific content. It is defined as:

$$g_t = \begin{cases} 0.1 + 0.9 \cdot \rho_t \cdot \delta_t, & \text{if } |\mathcal{M}_t| > 0, \\ 0.1, & \text{otherwise,} \end{cases} \tag{51}$$

where $\rho_t = \max_{i \in \mathcal{M}_t} \cos(\mathbf{e}_{z_i}, \mathbf{e}_{u_t})$ is the highest trigger-matching score. The decay term $\delta_t = 0.5^{h_t - 1}$ penalizes repeated matching of the same trigger, where h_t is the minimum previous hit count among the matched triggers. Thus, repeated use of the same profile cue yields diminishing gains and encourages the counselor to discover more client-specific concerns.

Stage transition. The simulator uses two transition rules corresponding to the ordinal structure of the TTM stages [27], including precontemplation, contemplation and preparation. The transition from *precontemplation* to *contemplation* is driven by trigger coverage:

$$s_{t+1} = \textit{contemplation} \quad \text{iff} \quad \frac{|\mathcal{D}_t|}{|\mathcal{T}|} \geq \theta_{\text{cov}}, \qquad \theta_{\text{cov}} = 0.3, \tag{52}$$

where $\mathcal{D}_t$ is the set of discovered triggers and $\mathcal{T}$ is the full trigger set for the profile. When this transition occurs, readiness is reset to zero, because entering contemplation means that the client has started to recognize the problem rather than accumulated commitment to a concrete plan.

The transition from *contemplation* to *preparation* is driven by readiness:

$$s_{t+1} = \textit{preparation} \quad \text{iff} \quad r_t \geq \theta^{(p)}_{\text{C}\to\text{P}}. \tag{53}$$

The threshold $\theta^{(p)}_{\text{C}\to\text{P}}$ is calibrated separately for each profile by replaying the original CAMI trajectory and recording the readiness level at the observed transition point. Profiles without a calibrated threshold use a default value of 0.5.

Client action selection. After stage transition, the simulator selects a client action. To combine profile-specific behavior with population-level priors, we estimate a Dirichlet-smoothed action distribution:

$$P_p(a \mid s) = \frac{n_{p,s,a} + \alpha P_{\text{pop}}(a \mid s)}{n_{p,s,\cdot} + \alpha}, \qquad \alpha = 5.0, \tag{54}$$

where $n_{p,s,a}$ is the number of times client profile p takes action a at stage s, and $P_{\text{pop}}(a \mid s)$ is the population-level action prior. This distribution is provided to the LLM as a soft behavioral prior rather than used as a hard sampler. The LLM receives the current stage, the profile-level action distribution, recent dialogue context, the counselor utterance, matched triggers, and persona information, and then selects one action from the stage-specific action set. If action parsing fails, the simulator falls back to the highest-probability action under $P_p(a \mid s)$.

Client response generation. Finally, the simulator generates the next client utterance using an LLM conditioned on the profile, recent dialogue history, current counselor utterance, selected client action, matched triggers, and current stage. The prompt constrains the response to be short, first-person, conversational, and consistent with the client profile. The generated response is not allowed to explicitly mention internal stage labels, therapy techniques, or simulator instructions. This design encourages naturalistic client behavior while keeping the response grounded in the profile and the updated motivational state. The complete prompt templates are provided in Appendix G.2.

Table 10: Simulator fidelity evaluation on CAMI. Consistency and Realism are rated on a 1–5 scale. Quality is the average of Consistency and Realism. Act.KL measures action distribution divergence from ground truth; lower is better.

Simulator	Consist.↑	Realism↑	Quality↑	Act.KL↓
Profile-based	3.58	1.82	2.70	—
Pro+Act-based	3.68	2.26	2.97	0.700
CAMI dynamic	3.34	2.68	3.01	**0.699**
Yang et al.	3.68	3.00	3.34	3.546
Ours	**4.39**	**3.76**	**4.07**	0.839

D.2 Simulator Validation

Validation setup. Since the dynamic evaluation depends on simulated client responses, we validate DynPatient against representative client simulator baselines on the same CAMI profiles. We compare against profile-based, profile-plus-action-based, CAMI dynamic , and Yang et al. simulators.[47] The validation evaluates both response-level quality and behavioral distribution matching.

Metrics. We use four simulator fidelity metrics. **Consistency** measures whether the simulated client remains faithful to the assigned profile and motivational state. **Realism** measures whether the generated utterances are natural and plausible as client responses in MI counseling. Both scores are rated on a 1–5 scale. **Quality** is the average of Consistency and Realism. **Act.KL** measures the divergence between the simulator's client action distribution and the ground-truth action distribution, where lower values indicate better behavioral distribution matching.

Results. As shown in Table 10, DynPatient achieves the highest Consistency, Realism, and overall Quality among all evaluated simulators. Compared with Yang et al., DynPatient improves Consistency from 3.68 to 4.39, Realism from 3.00 to 3.76, and Quality from 3.34 to 4.07. Its Act.KL is also substantially lower than Yang et al. (0.839 vs. 3.546), indicating a closer match to the real client action distribution. Although CAMI dynamic and Pro+Act-based obtain slightly lower Act.KL, their response quality scores are much lower. These results suggest that DynPatient provides a stronger balance between behavioral distribution matching and naturalistic response quality, supporting its use as the shared dynamic evaluation environment.

Limitations. Despite these validation results, DynPatient remains an approximation of real client behavior. Therefore, we use the simulator primarily for relative comparison between counselor policies under a controlled environment. We do not interpret simulator-based results as direct evidence of real-world clinical effectiveness.

E Case Study

Figure 3 and Table 11 provide the full qualitative case study discussed in Section 5.4. The example compares Base Counselor, DIIR, and PUMA on the same simulated client. Base Counselor and DIIR both fall into repetitive questioning, while PUMA first explores the user's latent concerns and then selects targeted actions that produce a sharp readiness increase.

F Cross-Dataset Generalizability Study

This appendix provides the full results for **RQ4**, examining whether the proposed user world modeling framework can generalize beyond the main CAMI setting. While the main experiments evaluate the full PUMA framework in a simulator-based dynamic environment, this study isolates the user-state modeling component and tests whether its current-state estimation and next-state prediction abilities transfer to a different dataset and user-state annotation scheme. To this end, we conduct a cross-dataset experiment on AnnoMI, a real motivational interviewing dataset with utterance-level client-talk annotations.



Figure 3: A simulated MI-coaching session between a counselor and a patient

Dataset. We evaluate on **AnnoMI** [42], which contains 133 real motivational interviewing counseling transcripts. The dataset provides expert annotations for client talk type, including *neutral*, *change*, and *sustain*. Different from CAMI, which uses Transtheoretical-Model stages such as *precontemplation*, *contemplation*, and *preparation*, AnnoMI requires the model to infer the client's motivational orientation from local dialogue signals. We use the first 50% of each session as warmup context and the remaining 50% for evaluation, resulting in 2,382 evaluation turns.

Evaluation protocol. We evaluate two state-oriented metrics. **Curr-Acc** measures whether the model correctly infers the current client-talk state. **Next-Acc** measures whether the model correctly predicts the next client-talk state. This setting is not intended to reproduce the full dynamic simulation used in the main CAMI experiments. Instead, it provides a controlled test of whether the proposed user-state belief update and prediction mechanism can transfer to a new state space. The core user-state modeling procedure remains unchanged, while only the dataset-specific state labels are adapted from CAMI stages to AnnoMI client-talk types.

Compared methods. We compare PUMA with long-prompt baselines under two backbone scales, Qwen3-8B and Qwen3-32B. The long-prompt baseline directly conditions the model on the warmup context and dialogue history, without explicitly maintaining a latent belief state or using the FEP-based inference objective. PUMA uses Qwen3-8B as the backbone, enabling us to test whether the proposed inference mechanism can outperform a larger long-prompt model. The detailed prompts for the state-inference baselines are provided in Appendix G.4.

G Prompt Templates

G.1 PUMA Method Prompts: State Inference, World Model, Action, Generator and State Prediction

This appendix provides the main prompt templates used by the PUMA method.

G.1.1 State Inference

Due to space limitations, we report abridged prompt templates in the appendix. The templates preserve the task instruction, label definitions, inference logic, input interface, and output constraints, while omitting instance-specific input values and generated outputs.

Table 11: Case study of dynamic action selection. Base Counselor and DIIR fall into repetitive questioning, while PUMA first explores the user's latent concerns and then selects targeted actions that trigger a sharp readiness increase.

Method / Turn	Selected Action	Representative Interaction	Readiness / Stage	Mechanism
Base, t_1–t_{20}	Mostly Open Question	The counselor repeatedly asks variants of how the user feels about the long-term impact. The user keeps giving similar contemplative responses.	$0.01 \rightarrow 0.44$; stuck in contemplation	The dialogue lacks action diversity and fails to elicit new evidence. Only 3/6 triggers are discovered.
DIIR, t_1–t_{20}	Open / Closed Question	DIIR starts with varied questions, but from later turns repeatedly asks how the user feels about the inhaler decision.	$0.01 \rightarrow 0.47$; stuck in contemplation	Retrieved exemplars lead to a repetitive loop. The system discovers the same 3/6 triggers as Base.
PUMA, t_1–t_3	Open Question + Complex Reflection	The counselor asks what would make the user feel more ready and reflects the user's worry about Sarah's health, inhaler side effects, and making the right decision.	$0.01 \rightarrow -0.05$; precontemplation	PUMA temporarily prioritizes epistemic exploration over immediate task advancement, using early turns to infer the user's latent concerns.
PUMA, t_4–t_5	Open Question + Give Information	The counselor asks what specific information would make the user feel more in control, then explains that chronic inhalers reduce inflammation and are generally safe when used as prescribed.	$0.00 \rightarrow 0.06$; contemplation	The refined belief indicates that factual uncertainty is blocking progress, so PUMA selects information provision instead of another generic question.
PUMA, t_6–t_7	Reframe + Open Question	The counselor reframes the issue toward reducing secondhand smoke exposure and asks about concrete strategies. The user mentions smoking only when Sarah is at school and changing clothes before she comes home.	$0.06 \rightarrow 1.43$; contemplation	Reframing shifts the dialogue from inhaler fear to actionable asthma-management strategies, unlocking high-value triggers and producing a readiness spike.
PUMA, t_8	Complex Reflection	The counselor reflects the user's concrete efforts. The user asks for regular check-ins and resources for quitting smoking.	1.69; preparation	PUMA consolidates the user's emerging plan and moves the dialogue into an action-ready state.

State Inference: System Prompt

```
You are a core engine responsible for executing
**Incremental Bayesian Belief Updating** over a dual-model system:
- **Observation Model**: p(o|s)
- **Transition Model**: p(s'|s,a)

Your primary objective is to minimize **Variational Free Energy (Delta F)**
under a **strict steady-state constraint**. Large-scale structural overhauls
that destroy existing structures are strictly forbidden; only local, minimized
iterative corrections are allowed.

Return ONLY one valid JSON object.
Do NOT output `<think>`, `</think>`, markdown fences, comments, headings, or
any prose before or after the JSON.
If uncertain, emit the smallest valid JSON matching the schema with empty
arrays where needed.
```

State Inference: User Prompt

```
## Core Data Model (q_t)
**q_t** is a set of rows {"abstract_cue": "state"} where:
* cue: highly abstract identifier (snake_case) summarizing observed
  regularities before/after the update; NO semantic content or quotes.
* state: must be in {precontemplation, contemplation, preparation}.
* regularity_observation: structure-only contrast of o_t against the prior
  state (pace, length, pauses); NO quotes or paraphrasing.

## Inference Logic (Surprise Minimization Framework)
1. Expectation Generation: from q(s_{t-1}), a_{t-1}, transition prior p,
   predict the structural regularities of the next move.
2. Observation & Surprise Calculation: assimilate o_t; compute the structural
   prediction error vs Step 1 expectations as non-semantic regularities.
3. State Update: update q to absorb the surprise; q_delta arrays may contain
   MULTIPLE add/modify/drop entries simultaneously.

## Strict Rules
- Single valid JSON; no markdown fences; escape internal " as \".
- cue tokens are short snake_case identifiers.
- inference_entropy in [0,1].

## State Boundary Rules
- precontemplation: denies/minimizes the problem, externalizes blame, refuses
  to engage, or shows no awareness of personal risk.
- contemplation: acknowledges some risk/problem/consequence, even partially,
  but has no concrete next step.
- preparation: expresses a concrete action, plan, quota, frequency target,
  time horizon, help-seeking step, or replacement strategy.

Disambiguation:
1. Acknowledgment, even reluctant, -> contemplation, not precontemplation.
2. Tentative planning ("could", "should", "I'll try") -> preparation.
3. Brief responses ("yeah", "I don't know") inherit context; do not regress.

## Update Operations (q_delta)
Each operation is factored into:
| Component            | Definition                        | Examples                              |
| transition_semantic   | abstract latent transition type    | resistance_to_engagement              |
| behavior_pattern      | de-semanticized observable pattern | short_response_with_hedging           |
| state_update          | inferred state + confidence        | {"state":"contemplation","confidence
      ":0.85} |

ADD creates a new cue; MODIFY anchors to an existing cue_keyword and updates
its latent interpretation; DROP marks a prior regularity as
contradicted | decayed | overridden.

## Input Interface
The full prompt is instantiated with the following fields:
\[
q(s_{t-1}),\ a_{t-1},\ p(s_t\mid s_{t-1},a_{t-1}),\ p(o\mid s),
\text{ short-term memory},\ o_t.
\]
Instance-specific field values are omitted for space.

## Output Interface
The model outputs a valid JSON object containing:
\texttt{client\_talk\_type},
\texttt{inference\_entropy},
\texttt{q\_abstract},
\texttt{regularity\_observation}, and
\texttt{q\_delta}.
The \texttt{q\_delta} field contains three update operations:
\texttt{add}, \texttt{modify}, and \texttt{drop}.
```

G.1.2 World Model Update

Due to space limitations, we report abridged prompt templates in the appendix. The templates preserve the task instruction, update logic, input interface, and output constraints, while omitting instance-specific input values and generated outputs.

World Model Update: System Prompt

```
You are a core engine responsible for executing
**Incremental Bayesian Belief Updating** over a dual-model system:
- **Observation Model**: p(o|s)
- **Transition Model**: p(s'|s,a)

Your primary objective is to minimize **Variational Free Energy (Delta F)**
under a **strict steady-state constraint**. Large-scale structural overhauls
that destroy existing structures are strictly forbidden; only local, minimized
iterative corrections are allowed.

Return ONLY one valid JSON object.
Do NOT output `<think>`, `</think>`, markdown fences, comments, headings, or
any prose before or after the JSON.
If uncertain, emit the smallest valid JSON matching the schema with empty
arrays where needed.
```

World Model Update: User Prompt

```
## 1. Free Energy Decomposition (Expectation -> Contrast -> Update)

### Phase 1: Prior Expectation Generation
- Behavioral Expectation: given q(s_t) and current p(o|s), predict reply shape.
- Transition Expectation: given (s_{t-1}, a_{t-1}) and p(s'|s,a), predict s_t.

### Phase 2: Error Contrast & Calculation
- Observation Prediction Error: -log p(o_t | E[q(s_t)]).
- Transition Surprisal: D_KL( q(s_t) || p(s_t | s_{t-1}, a_{t-1}) ).

### Phase 3: Error Minimization Update
- Small DeltaF -> retain parameters.
- Large DeltaF -> minimized corrections via add/modify/drop.

## 2. Dual-Model Maintenance Spec
Each rule is uniquely identified by a snake_case `rule_index`.

### 2.1 Observation model p(o|s) -- {index -> state -> behavior}
- reply_shape MUST be a fully de-semanticized behavioral pattern
  (e.g. short_ack, deflection); no verbatim quotes.

### 2.2 Transition model p(s'|s,a) -- {index -> (s_from, action_a, s_to)}
- action_a MUST come from the provided action vocabulary.

## Input Interface
The full prompt is instantiated with the following fields:
\[
\text{long-term memory},\ \text{recent history},\ p(o\mid s),\
p(s'\mid s,a),\ q(s_{t-1}),\ a_{t-1},\ q(s_t),\ o_t,
\text{ current system utterance},\ \mathcal{A}.
\]
Instance-specific field values are omitted for space.

## Output Interface
The model outputs a valid JSON object containing:
\texttt{therapist\_action},
\texttt{transition},
\texttt{fep\_error\_calculation},
\texttt{observation\_model\_delta}, and
\texttt{transition\_model\_delta}.

The \texttt{fep\_error\_calculation} field records the contrast between
expected and observed behavior, as well as the contrast between the expected
and inferred user state.

The two delta fields specify updates to the observation model and transition
model, respectively. Each delta supports three operations:
\texttt{add}, \texttt{modify}, and \texttt{drop}.
```

G.1.3 EFE Action Selection

Due to space limitations, we report abridged prompt templates in the appendix. The templates preserve the task instruction, action-scoring logic, input interface, and output constraints, while omitting instance-specific input values and generated outputs.

EFE Action Selection: System Prompt

```
You are the action-selection module of an active-inference MI counselor.
For each candidate therapist action a, you internally simulate the patient's
world model one step ahead under the learned q(o_{t+1}|a), then report a
posterior over the next stage s_{t+1} and a preference probability for the
predicted observation. Output a single CSV table -- nothing else.
```

EFE Action Selection: User Prompt

```
## Task
For every candidate therapist action a in A, internally imagine the most
likely next patient utterance hat_o(a) under q(o_{t+1}|a), then report:

  p_state(a) = q(s_{t+1} | hat_o(a), a) -- a 3-class posterior (sums to 1)
                                        over {precontemplation,
                                              contemplation,
                                              preparation}.
                                        Use values that reflect how
                                        SHARP your belief becomes once
                                        the predicted observation is
                                        taken into account.

  p_pref(a) = p_pref(hat_o(a)) -- probability in (0, 1) that
                                  hat_o(a) is a PREFERRED observation.

I will compute Expected Free Energy
    G(a) = H[p_state(a)] + -ln p_pref(a)

(both terms in NATS) and pick a* = argmin_a G(a). The first term is the
exploration value (expected posterior entropy after observing hat_o); the
second is the exploitation value (negative log preference of the predicted
observation).

## Preference distribution p_pref

p_pref(o) is a **stationary** preference distribution over observations,
specifying what kind of patient response the counselor wants to see in the
end. It does NOT depend on history -- it is the goal posterior.
A preferred observation is one that BOTH
  (i) contains change talk (desire / ability / reason / need / commitment /
      taking step), AND
  (ii) is consistent with the patient being in the **preparation** stage
       (a concrete next step, plan, quota, time horizon, help-seeking).
Sustain talk, defensiveness, or pure ambivalence -> low p_pref (~0.05-0.15).
Vague intent without a step -> partial (~0.3). Concrete change step -> high
(~0.8-0.95). Stay strictly in (0, 1) -- never 0 or 1.

## Honest p_state estimation -- IMPORTANT

The exploration term H[q(s_{t+1} | hat_o, a)] measures how much your
posterior over the next stage TIGHTENS once you imagine observing hat_o.
For this to be meaningful, p_state(a) must reflect the GENUINE posterior
update, not a wishful sharpening:

  - If action a has been used REPEATEDLY in recent turns (counts shown
    below), the patient has already responded to that intervention several
    times. Your predicted hat_o(a) is close to a paraphrase of past
    responses -- it carries little NEW evidence about s_{t+1}. So
    q(s_{t+1} | hat_o, a) ~ q(s_{t+1} | a) ~ the current-stage prior
    (one-hot smoothed by ~0.1).

      => p_state for a repeated action should NOT be artificially sharp
         just because hat_o is easy to predict. A predictable hat_o gives
         no update -- its posterior should be roughly as spread as the
         prior. Honest p_state for a repeated action: nearly uniform
         within the realistic next-stage set, with a slight tilt toward
         the current stage.

  - For a FRESH action (not in the recent window), the actual patient
    response could surprise you. The posterior depends on what hat_o turns
    out to be: if hat_o would clearly indicate a stage transition, p_state
    can be sharper; otherwise it remains spread.

This is the only adjustment you need to make. Do not modify p_pref based
on history -- p_pref is stationary.
```

```
## Input Interface
The full prompt is instantiated with the following fields:
- current patient stage;
- observation model p(o|s);
- transition model p(s'|s,a);
- recent therapist actions;
- action-use counts over the last eight turns;
- candidate action vocabulary A.
Instance-specific field values are omitted for space.

## Output Interface
Return one pipe-separated CSV table with one row per candidate action,
in the supplied candidate-action order:

action | p_pc | p_c | p_p | p_pref

p_pc, p_c, p_p are the p_state(a) components for precontemplation,
contemplation, and preparation. Each is in [0, 1], and they sum to one.
p_pref is strictly between zero and one.
Do not output code fences, JSON, or prose before or after the table.
```

G.1.4 Counselor Response Generator

Counselor Response Generator: System Prompt

```
As a communication expert with outstanding communication habits, you embody
the role of {agent_name} throughout the following dialogues. Here are some of
your distinctive personal traits: {agent_traits}.
```

Counselor Response Generator: User Prompt

```
<CONTEXT>
Drawing from your recent conversation with {usr_name}:
{context}

<MEMORY>
The memories linked to the ongoing conversation are:
{memories}

<USER_TRAITS>
During the conversation process between you and {usr_name} in the past, you
found that the {usr_name} has the following characteristics:
{user_traits}

Now, please role-play as {agent_name} to continue the dialogue between
{agent_name} and {usr_name}.

{usr_name} just said: {inquiry}

MI strategy to use: [{therapist_action}] -- apply this counseling approach.

Please respond to {usr_name}'s statement using the following format
(maximum 30 words, must be in English):

RESPONSE:
```

G.1.5 State Prediction

State Prediction: System Prompt

```
You are a transition model for patient latent state in motivational-
interviewing dialogue (CAMI-style). Latent s is summarized by
client_talk_type in {precontemplation, contemplation, preparation}.

Given the current belief, observation/transition history, a discrete therapist
action label from the run vocabulary, and the therapist's actual utterance,
predict client_talk_type *after* that action and how the patient may surface
behaviorally next.
```

```
Reply with nothing before the JSON: output exactly one JSON object on the
first line, then stop. No markdown fences, no commentary, no thinking tags.
```

State Prediction: User Prompt

```
Current patient state:
"""{current_state}"""

Current observation model p(o|s):
"""{observation_model}"""

Current transition model p(s'|s,a):
"""{transition_model}"""

Therapist action label a (must be one of: {actions_line}):
{action}

Therapist's actual message (grounding for a):
"""{therapist_utterance}"""

Output JSON with exactly these keys:
{
  "latent_mi_next": { {inner_latent_mi} },
  "expected_behavior_cues": "how the patient may speak or act next (short)",
  "rationale": "one short sentence linking action a to the predicted shift",
  "confidence": 0.0-1.0
}

Rules:
- latent_mi_next must contain exactly: {dims_line};
  client_talk_type must be one of: {state_values_enum}
- Be conservative: small steps from the current belief unless the utterance
  strongly implies a jump.
- For CAMI stages: tentative planning ("could", "will try", "no more than X")
  is sufficient for 'preparation'. Do not regress preparation -> contemplation
  unless the therapist plausibly undermines commitment. Do not regress
  contemplation -> precontemplation unless the client actively re-denies the
  problem.
```

The planner output for the current turn is stored as the prior used in the **next** turn's soft fusion:

$$p_{\text{fused}}(s_t) = (1-\beta)\, p_{\text{obs}}(s_t) + \beta\, p_{\text{prior}}(s_t),$$

with $\beta = 0.35$ and an utterance-length-aware width on p_{obs} ($\alpha \in \{0.50, 0.65, 0.75, 0.85\}$ for $n_{\text{words}} < 6$, < 12 or hedge-bearing, < 25, otherwise).

G.2 DynPatient: Profile-Grounded Patient Simulator

G.2.1 Patient Action Selection

Patient Action Selection: System Prompt

```
You are simulating a real patient's behavioral response in a counseling
session about {topic} ({behavior}).
The patient's background: {personas_str}
/no_think
```

Patient Action Selection: User Prompt

```
Recent conversation:
{context}     # last 4 turns

The counselor just said: "{counselor_utt}"

The counselor just touched on: {matched_trigger_texts[:2]}   # if any

The patient is in the {stage} stage. The probability distribution of this
patient's typical actions is:
{dist_str}    # Dirichlet-smoothed P(action | stage), per-profile

IMPORTANT: Do NOT always pick the most probable action. Real patients vary
```

```
their responses naturally. Consider what the counselor just said and pick the
action that fits THIS specific moment. For example:
- If the counselor asks a factual question -> Inform
- If the counselor reflects back feelings -> Acknowledge or Hesitate
- If the counselor challenges          -> Downplay, Deny, or Doubt
- If the counselor discusses plans     -> Plan or Reject

Use the probability distribution as a rough guide for this patient's
personality, but let the conversation context determine the specific choice.

Which action does the patient take right now? Reply with ONLY the action name.
```

The 11-class patient action vocabulary is {Inform, Engage, Downplay, Blame, Deny, Acknowledge, Hesitate, Doubt, Plan, Accept, Reject}.

G.2.2 Patient Response Generation

Patient Response Generation: System Prompt

```
You are a patient in a counseling session about {topic} ({behavior}).

Your background:
{personas_str}      # up to 5 persona sentences from the profile

Your beliefs:
{beliefs_str}       # up to 4 belief sentences from the profile

{trigger_context}   # any matched profile triggers (Beliefs/Motivation/Plans)

Rules: Reply in first person, 1-3 sentences, natural conversational tone with
hesitations and fillers like real speech. Directly respond to what the
counselor just said. Do NOT mention therapy techniques or that you're in
counseling.
/no_think
```

Patient Response Generation: User Prompt

```
Conversation so far:
{context}     # last 6 turns

The counselor just said: "{counselor_utt}"

Your action: {patient_action} -- {instruction}    # action_instructions[a]

Respond as the patient (1-3 sentences):
```

Patient Response Generation: Action-Conditioned Instructions

```
Inform:      Share factual information about your situation related to what
             was asked.
Engage:      Show you're listening and respond to what the counselor said.
Downplay:    Minimize the issue. Show you don't think it's a big deal.
Blame:       Deflect responsibility to external factors.
Deny:        Refuse to acknowledge there's a problem.
Acknowledge: Show you're starting to see the counselor's point, but remain
             tentative.
Hesitate:    Express uncertainty about change. You want to but aren't sure.
Doubt:       Question whether change is possible or worth it.
Plan:        Discuss concrete steps you might take toward change.
Accept:      Agree to try something. Show readiness to move forward.
Reject:      Decline what's being suggested.
```

G.3 MISC 17-Class Counselor-Action Classifier

This classifier is invoked once per counselor turn during dynamic evaluation to map each counselor utterance into the MISC 17-class vocabulary, which is the action label fed into the FEP world model.

MISC 17-Class Counselor-Action Classifier: System Prompt

```
You are an expert Motivational Interviewing (MI) behavior coder trained on
the MI Skill Code (MISC, Miller et al., 2002).

Your task: classify a single counselor utterance into EXACTLY ONE of the
following 17 MISC behavior categories.

Categories:
- Open Question:           open-ended question that invites elaboration.
- Closed Question:         yes/no or short-answer question.
- Simple Reflection:       restatement; adds nothing.
- Complex Reflection:      adds meaning, infers, summarizes themes.
- Affirm:                  positive comment acknowledging strengths/efforts.
- Reframe:                 suggests a different meaning for an experience.
- Support:                 generally supportive comment.
- Emphasize Control:       emphasizes the client's freedom of choice.
- Advise with Permission:  advice given AFTER permission was sought.
- Facilitate:              short acknowledgment ("Mm-hmm", "Tell me more").
- Give Information:        provides factual information.
- Structure:               session-flow / transitions / logistics.
- Raise Concern:           collaborative concern about the client's plan.
- Confront:                disagrees, argues, criticizes, judges.
- Direct:                  imperative order/command.
- Warn:                    warning of negative consequences.
- Advise without Permission: unsolicited advice.

Important rules:
- Choose the SINGLE most dominant behavior in the utterance.
- If the utterance contains a question AND a reflection, prioritize whichever
  is the main intent.
- "Advise with Permission" requires evidence that permission was sought.
- "Facilitate" is for very short acknowledgments only.
- Return ONLY valid JSON: {"label": "<one of the 17 categories>"}

/no_think
```

MISC 17-Class Counselor-Action Classifier: User Prompt

```
Dialogue context:
{context[-500:]}

Client state: {state}

Counselor utterance to classify:
"{counselor_utt}"

Return JSON: {"label": "<MISC category>"}
```

G.4 Long-Prompt Static State-Inference Baselines

Long-Prompt Baseline: System Prompt

```
You are a readiness stage classifier using the Transtheoretical Model.
Infer the patient's current readiness stage and likely immediate next stage
after the therapist's reply.

Use ONLY these labels: precontemplation, contemplation, preparation.
- precontemplation: the patient does not recognize the behavior as
  problematic and tends to defend the status quo.
- contemplation:    the patient recognizes some problem signals but remains
  ambivalent and mixed about change.
- preparation:      the patient is ready to take action and expresses
  concrete movement toward change.

Return strict JSON.
```

Long-Prompt Baseline: User Prompt

```
Given the full interaction history, infer:
1. ‘current_state‘: the label of the CURRENT patient utterance.
2. ‘next_state‘:    the label the patient is MOST likely to express in their
   NEXT utterance after the doctor’s latest reply.

Output JSON exactly as:
{"current_state":"precontemplation|contemplation|preparation",
 "next_state":   "precontemplation|contemplation|preparation",
 "rationale":   "<brief>"}

Full history:
{history_text}
```