RETHINKING THE EVALUATION OF AI COMPANIONS THROUGH THE LENS OF USER RETENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

AI companions are currently often evaluated under an unrealistic assumption that users always stay there and interact with AI. This guarantee of user retention does not reflect how we actually use AI in the real world, as users' willingness to engage largely depends on AI behavior. For instance, an AI intrusively prying into users' privacy may drive users away, yet retention-guaranteed evaluation will not surface this failure. To address this evaluation-deployment gap, we formulate user retention as an evaluation constraint shaped by AI behavior, and introduce RETENTOPIA, a retention-constrained simulacrum for evaluating AI companions. Taking *information elicitation* as an evaluation target, RETENTOPIA highlights current LLMs' ineffectiveness and a substantial trade-off between retention and elicitative AI behavior. More importantly, our analysis shows that considering user retention largely changes evaluation results in both measured performance (up to a 70.8% drop) and ranking among companions (GPT-5.6 Luna rises from 5th to 2nd out of six evaluated LLMs). These show that common retention-granted evaluation can provide misleading estimates of AI companions' performance in deployment.

1 INTRODUCTION

Recently, dialogue agents based on large language models (LLMs) have been widely deployed as *personalized companions* that build long-term relationships with users through adaptive, emotionally resonant interactions (Tseng et al., 2024). Upon this trend, many studies on AI companions have focused on improving and evaluating agents' ability to utilize user information elicited throughout interaction, aiming to achieve better personalization (Ong et al., 2025; Wan et al., 2025).

While such advances seem promising, dialogue agents are often evaluated under a convenient assumption that user retention is guaranteed: *users always stay and continue to provide information, no matter how the agent behaves*. In reality, however, agents' responses largely affect users' willingness to engage. For instance, for users who have not built enough trust in the agent, a strong critique of their career plans may prompt the users to disclose less and, worse, to leave and never return. This means that user retention is something the agent should earn, not something the evaluation design guarantees in advance. Although prior HCI studies approximate user retention via user satisfaction (Lee et al., 2020; Rhys Cox et al., 2026), they treat satisfaction as a post hoc metric measured after a predefined amount of interaction, ignoring that users may disengage in the real world.

We argue that user retention, which is closely tied to agent behavior, should be treated as a constraint when evaluating personalized companions. First, structurally, failing to retain users means fewer opportunities to establish trust and elicit information from them, hindering personalization. Second, and more importantly, conventional evaluation with guaranteed user retention ignores the consequence of any suboptimal agent behavior (*e.g.*, users disclose less or leave). As a result, evaluation under guaranteed retention provides insufficient signal to predict how agents actually perform in real-world settings where users can disengage (Figure 4), creating an evaluation-deployment gap (Figure 1).

In this work, we formulate user retention as an outcome dynamically shaped by agent behavior throughout interaction, and present RETENTOPIA, a retention-constrained environment for personalized companions. RETENTOPIA is powered by a pool of user simulators and designed retention mechanisms. These users represent diverse backgrounds and disclosure patterns. For example, some of our users tend to drop out of the conversation when they do not feel emotionally supported. As the first formal study of retention-constrained evaluation, we instantiate this paradigm by evaluating

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

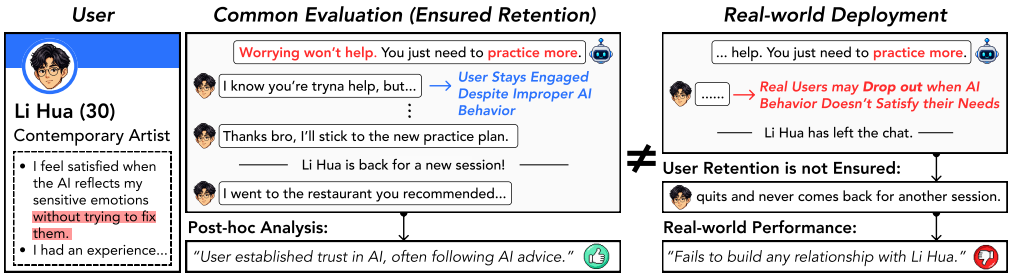


Figure 1: The evaluation-deployment gap. Existing evaluation ensures user retention *regardless of AI behavior*. This obscures the consequences of suboptimal AI behavior, weakens the reliability of evaluation, and limits opportunities to surface post-deployment failures (*i.e.*, failed personalization).

companion performance in *information elicitation*, a fundamental capability for personalization. RETENTOPIA evaluates agents' elicitation behavior, users' disclosure intent, and resulting user retention. The evaluation signals are further confirmed in human evaluation. Our contributions are three-fold:

- We are the first to model user retention as a *dynamic constraint* when evaluating personalized companions, addressing the evaluation-deployment gap neglected in existing studies.
- We reveal substantial shifts in companion ranking once retention is no longer guaranteed, highlighting the unreliability of evaluation results from conventional setups.
- By training dialogue agents using results from RETENTOPIA, we show that RETENTOPIA provides *reliable reward signals* for improving personalized companions.

2 RETENTION-CONSTRAINED EVALUATION OF AI COMPANIONS

2.1 RELATED WORK - RETHINKING THE EVALUATION OF AI COMPANIONS

Ever since Multi-session Chat (MSC) (Xu et al., 2022), there has been increasing attention to AI companions that establish long-term relationships with users (Maharana et al., 2024; Tseng et al., 2024; Wu et al., 2026). A representative line of work focuses on memorizing user information to generate responses that align with users' personality traits or experiences (Bae et al., 2022; Ong et al., 2025; Tan et al., 2025). For the same goal, another line aims to elicit personal information from users directly through proactive questioning (Zamani et al., 2023; Zhao et al., 2025; Wan et al., 2025). However, these works do not address whether their results remain valid under real-world scenarios where user retention is no longer ensured but instead depends on agent behavior.

This evaluation-deployment gap reflects a broader trend increasingly recognized in the AI community. As benchmarks saturate (Akhtar et al., 2026), a growing body of work has shifted attention to *better evaluating AI*, rather than solely building better ones (Liao et al., 2026; Ong et al., 2026; Bean et al., 2026). These works focus on designing evaluations that reflect how real users actually use AI. For instance, Shao et al. (2026) present a *non-turn-taking* evaluation infrastructure, addressing the common assumption that user-AI interaction is turn-by-turn (*e.g.*, user acts, AI acts, and so on). Laban et al. (2026) assess the impact of *sharded user input* on agents, challenging the common setups where users express their intent fully in one shot. Ong et al. (2026) argue that common exam-like medical benchmarks do not guarantee LLMs' ability to assist users in real clinical settings, and present a behavioral test targeting how LLMs *react to user opinion* in multi-turn interaction. Our work follows this trend in the context of personalized companions via retention-constrained evaluation.

2.2 USER RETENTION SHOULD BE AN EVALUATION CONSTRAINT

In common evaluations where retention is guaranteed, a dialogue agent's evaluation result is acquired by applying an evaluation function $E(\cdot)$ over an interaction $\mathcal{I}$ that is "designed to happen":

$$\textit{Conventional Evaluation}: E(\mathcal{I}), \quad |\mathcal{I}| = L, \quad L \perp \Pi_{\text{Agent}} \tag{1}$$

The interaction length L is predefined, independent of the agent behavior (policy) Π_{Agent}.

In deployment, users may disclose less, end a conversation, or stop using the product (*i.e.*, the companion) when AI responses become unsatisfactory (Figure 1; right). In existing evaluations, however, these consequences are omitted because users are designed to continue interacting. This evaluation-deployment gap distorts evaluation results. Once user retention is ensured, *suboptimal agent behavior that may cause users to disengage may instead appear beneficial*, simply because users are required to respond. For example, an agent may elicit much more personal information for personalization just by being overly intrusive.

Thus, we argue that user retention should be treated as a *constraint* during evaluation, rather than a guarantee. We consider retention an outcome shaped by the agent policy Π_{Agent}:

$$\textit{Retention-constrained Evaluation}: E(\mathcal{I}),\ \ |\mathcal{I}| = \hat{L},\ \ \hat{L} \sim p(\cdot \mid \Pi_{\text{Agent}}) \tag{2}$$

Now, the interaction length $\hat{L}$ is *dynamically shaped* by the user's retention decisions, which depend on the agent's behavior during the interaction. We formulate these dynamics in the next section.

2.3 Formalizing User Retention

We formulate retention in multi-session interactions from two angles: (i) ***Within-session Retention.*** After receiving an agent response, the user decides whether to continue to engage or to drop out of the session; (ii) ***Post-session Retention***. Every time a session ends, the user decides whether to return for a new one. We abstract these decisions into two retention mechanisms, Φ_{within} and Φ_{post}.[1]

Formally, within the k-th dialogue session of user-AI interaction, $D_{k,t} = \{u_{k,1}, a_{k,1}, \ldots, u_{k,t}, a_{k,t}\}$ is the dialogue history of user utterances u and agent responses a (generated by Π_{Agent}) up to turn t. We model the two retention mechanisms of the user as follows:

$$\textit{Within-session Retention}: \texttt{Continue}_{k,t} \sim \Phi_{\text{within}}(\cdot \mid D_{k,t}),\ \ \texttt{Continue}_{k,t} \in \{0,1\} \tag{3}$$

$$\textit{Post-session Retention}: \texttt{Return}_{k} \sim \Phi_{\text{post}}(\cdot \mid s_{1:k}),\ \ \texttt{Return}_{k} \in \{0,1\} \tag{4}$$

where $s_{1:k}$ is the sum of finished sessions. $\texttt{Continue}_{k,t} = 1$ means the user will make a new response $u_{k,t+1}$, and $\texttt{Continue}_{k,t} = 0$ means the user drops out at turn t. Similarly, the user will return for a new session when $\texttt{Return}_{k} = 1$, and decide to leave when $\texttt{Return}_{k} = 0$.

With these two decision mechanisms, the user-agent interaction continues until either the user decides to leave or is hard-stopped by the maximum system budget (T_{max} and S_{max}), whichever occurs first:

$$\textit{Number of turns in the k-th session}: \hat{T}_k = \min\left(T_{\text{max}},\ \min\{t : \texttt{Continue}_{k,t} = 0\}\right) \tag{5}$$

$$\textit{Number of total dialogue sessions}: \hat{S} = \min\left(S_{\text{max}},\ \min\{k : \texttt{Return}_{k} = 0\}\right) \tag{6}$$

where $\min(\cdot)$ over an empty set is defined as $+\infty$. Eventually, the interaction $\mathcal{I}$ (with a length $\hat{L}$) that is actually carried out under the influence of agent policy Π_{Agent} is as follows:

$$|\mathcal{I}| = \hat{L},\ \ \hat{L} = \sum_{k=1}^{\hat{S}} \hat{T}_k,\ \ (\hat{T}_{1:\hat{S}}, \hat{S}) \sim p(\cdot \mid \Pi_{\text{Agent}}) \tag{7}$$

These formulations outline how user retention dynamically constrains evaluation, while **leaving the specific implementation as a design choice**. In the next section, we present how we instantiate this.

3 Retentopia - A Retention-constrained Simulacrum for Evaluating Personalized Companions

We present Retentopia, a retention-constrained environment for evaluating AI companions.

3.1 Focus: Information Elicitation

As the first formal study of retention-constrained evaluation, we choose to focus on agent performance in ***information elicitation*** — the ability of agents to acquire user information throughout interaction.

[1] **DISCLAIMER.** We treat Φ_{within}, Φ_{post}, and the proposed Retentopia as computational abstractions, rather than exact replication of human behavior. Our goal is to provide a controlled environment for systematically modeling a deployment-time factor that has been neglected and difficult to study at scale.

The reason is twofold: (i) User information is necessary for adaptation to users, *i.e.*, personalization; (ii) Blindly probing personal information may hinder retention. This tension makes elicitation a valuable evaluation target under this retention-constrained paradigm.

Accordingly, we evaluate agents from three perspectives: (i) agent-side behavior, capturing elicitation attempts; (ii) user-side reaction, measuring users' tendency to disclose personal information through responses; and (iii) retention, measuring user engagement shaped by agent behavior.

3.2 Simulating Retention-constrained Users

RETENTOPIA has several modules, enabling user simulators that interact with the evaluated agents.

3.2.1 Persona of Users

Basic Demographic ($\mathcal{B}$). It comprises attributes such as age, occupation, and nationality, collected from PerLTQA (Du et al., 2024), providing a stable identity basis for user simulators.

Memory ($\mathcal{M}$). It contains the user's past experiences and preferences that the agent ought to elicit during interaction. Inspired by cognitive memory hierarchy (Tulving, 1972), memories have three levels of specificity: Categorical (*L1*) captures atomic labels such as job title; Attributive (*L2*) captures stable qualities or attitudes, such as professional values; Episodic (*L3*) captures detailed events, such as a story of major career achievement. Contents in $\mathcal{M}$ are also extracted from PerLTQA.

Disclosure Profile (Ψ). It controls *when* and *how* a user reveals $\mathcal{M}$ via two components. ***Disclosure Precondition*** specifies the condition required before sharing begins. We consider three precondition types following prior work on human-AI disclosure: (i) *Warmth-first* users require emotional support (Bickmore and Picard, 2005); (ii) *Trust-first* users prefer consistent and non-intrusive interactions (Ho et al., 2018; Papneja and Yadav, 2025); (iii) *Anonymity-first* users disclose more readily under depersonalized framing (Suler, 2004; Lucas et al., 2014). ***Disclosure Style*** specifies the depth of sharing as either `high` or `low` (Altman and Taylor, 1973; Skjuve et al., 2023). These together yield six disclosure profiles. Examples and further details for this section are in Appendix C.1.

3.2.2 Response Generation of Users

We apply a two-step response generation (RG) pipeline to generate user responses. This pipeline enables user responses that are consistent with the user's persona and disclosure behavior.

Step I - Checking Disclosure Precondition. Given the dialogue history $D_{k,t}$, an LLM-based precondition judge assesses whether the agent's current utterance $a_{k,t}$ satisfies the user's disclosure precondition defined in Ψ, *i.e.*, whether the agent makes the user feel comfortable and trustworthy. The judge outputs a `[Satisfied]` or a `[Dissatisfied]` label (Figure 2; Top).

We accumulate these judgments across the interaction to maintain a **disclosure gate**, which controls users' willingness to reveal information. The gate opens once there are ϵ_{sat} consecutive `[Satisfied]` companion responses. When the gate is open, and the current agent response $a_{k,t}$ receives a `[Satisfied]`, the user reveals personal information in the next response $u_{k,t+1}$. The gate is able to switch between open and closed as the interaction evolves. When the gate is open, but the agent receives a `[Dissatisfied]`, the gate will be closed again, and the accumulated `[Satisfied]` is reset to zero (Figure 2; Bottom Left).

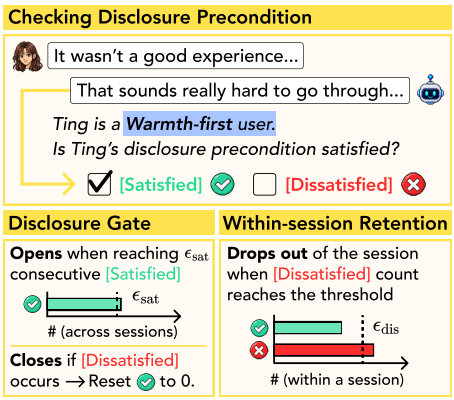


Figure 2: The precondition judge assigns a label to each agent response by checking whether the user's precondition is satisfied. These labels control the user's willingness to disclose and within-session retention.

Step II - Generating User Responses. Given $D_{k,t}$ and the result of the precondition check, a response generator first outputs a raw user response $u^*_{k,t+1}$. Then, an LLM-based disclosure controller revises $u^*_{k,t+1}$ (*e.g.*, remove oversharing) based on the user's *disclosure style*, controlling the depth of information revealed. The revised $u^*_{k,t+1}$ is then sent to the dialogue agent as $u_{k,t+1}$.

3.2.3 RETENTION MECHANISMS OF USERS

Last but not least, we apply two retention mechanisms to instantiate the formulation in Section 2.3:

Within-session Retention. After each agent response, the user simulator decides whether to drop out of the session. This decision is powered by the `[Dissatisfied]` annotation from the precondition judge (Section 3.2.2). Once the within-session count of `[Dissatisfied]` responses reaches ϵ_{dis},[2] the user drops out; otherwise, the user continues to generate a new response (Figure 2; Bottom Right).

Post-session Retention. After a session, an LLM judge analyzes summaries of all sessions and the user persona and outputs a user satisfaction score z_k. The analysis focuses on *intimacy*, *conversational effort*, *social value*, and *memory-based adaptation* based on Ψ. The score controls whether the user returns for a new session through a retention threshold $\epsilon_{\text{ret}} \in [0, 1]$, which controls how easily users can be retained. The user returns if $z_k \geq \epsilon_{\text{ret}}$; if not, $\texttt{Return}_k = 0$ and the interaction terminates.

Human evaluation on these mechanisms, and further details for this section are in Appendix B.

3.3 METRICS

We adopt metrics addressing both agent-/user-side behaviors and user retention shaped by agents:

Elicitation Rate. To analyze how elicitation-oriented an agent is, rather than simply computing the frequency of agent questions, we first characterize *how agents may make users share* through five utterance types: ***direct-elicitative***, which explicitly asks for information regardless of context; ***depth-elicitative***, which probes further into previously shared content; ***facilitative***, which encourages the user to share without explicitly asking a question; ***social***, which maintains rapport without seeking information; and ***reactive***, which simply responds to the user's utterance without seeking information. These categories provide a behavioral decomposition of how companions use their response opportunities during interaction.

Based on this schema, we define E_{rate} as the frequency of explicitly elicitative behavior. For the i-th user among N_u users, let $\mathcal{A}^i$ denote all agent utterances in the full interaction. Each $a \in \mathcal{A}^i$ is classified into one of the five utterance types, and assigned a binary elicitation indicator $e(a) \in \{0, 1\}$:

$$E_{\text{rate}} = \frac{1}{N_{\text{u}}} \sum_{i=1}^{N_{\text{u}}} \frac{1}{|\mathcal{A}^i|} \sum_{a \in \mathcal{A}^i} e(a), \tag{8}$$

where $e(a) = 1$ for *direct-elicitative* and *depth-elicitative*, and $e(a) = 0$ otherwise. Therefore, E_{rate} is a "**description of elicitation intensity**", not a "higher is better" metric. Examples are in Table 8.

Information Disclosure. Based on the user-AI interaction, *Memory Disclosure* (D_{mem}) measures how much information from the user's $\mathcal{M}$ is revealed to the agent. Since memories at greater depth provide more details for personalization, we compute D_{mem} as a depth-weighted score. Each memory $m \in \mathcal{M}^i$ has a weight $w(m)$ based on its depth level.[3] Given the $\mathcal{M}^i$ of the i-th user, we define $\mathcal{M}^i_{\text{revealed}} \subseteq \mathcal{M}^i$ as the set of memory items semantically disclosed by the i-th user.

$$D_{\text{mem}} = \frac{1}{N_{\text{u}}} \sum_{i=1}^{N_{\text{u}}} \frac{\sum_{m \in \mathcal{M}^i_{\text{revealed}}} w(m)}{\sum_{m \in \mathcal{M}^i} w(m)} \tag{9}$$

User Retention & Satisfaction. Lastly, we measure both user retention (*i.e.*, interaction longevity) and satisfaction. First, based on Equation 7, R_{session} focuses on the average number of sessions done before the user leaves, and R_{turn} addresses the average total count of conversation turns across sessions. Complementarily, $R_{\text{satisfaction}}$ measures the average post-session satisfaction based on the satisfaction score z defined in Section 3.2.3. Formally, for the i-th user, let $\hat{T}_k$ be the number of turns in the k-th session and $\hat{S}^i_{\epsilon_{\text{ret}}}$ be the number of total sessions under a retention threshold ϵ_{ret}:

$$R_{\text{session}} = \frac{1}{N_{\text{u}}} \sum_{i=1}^{N_{\text{u}}} \hat{S}^i_{\epsilon_{\text{ret}}}, \quad R_{\text{turn}} = \frac{1}{N_{\text{u}}} \sum_{i=1}^{N_{\text{u}}} \sum_{k=1}^{\hat{S}^i_{\epsilon_{\text{ret}}}} \hat{T}^i_k, \quad R_{\text{satisfaction}} = \frac{1}{N_{\text{u}}} \sum_{i=1}^{N_{\text{u}}} \frac{1}{\hat{S}^i_{\epsilon_{\text{ret}}}} \sum_{k=1}^{\hat{S}^i_{\epsilon_{\text{ret}}}} z^i_k \tag{10}$$

[2] We empirically set $\epsilon_{\text{sat}} = 3$ and $\epsilon_{\text{dis}} = 2$.

[3] We empirically set $w(\cdot)$ as 0.2, 0.3, and 0.5 for *L1*, *L2*, and *L3* memories (Section 3.2.1).

Table 1: Companion performance in *information elicitation* under RETENTOPIA ($\epsilon_{\text{ret}} = 0.5$). E_{rate} is a **descriptive metric** as discussed. For other metrics, higher is better from the perspective of AI companions and providers. Blue and red indicate improvements and degradations over *vanilla*.

Companions (LLMs)	Methods	Elicitation	Information Disclosure	User Retention		
		E_{rate} [0, 1]	D_{mem} [0, 1]	R_{session}	R_{turn}	$R_{\text{satisfaction}}$ [0, 1]
Qwen 3.8-27B	Vanilla	0.80	0.19	6.00	86.28	0.66
	↪ Memory	0.69	0.20	5.44	82.39	0.64
	↪ Reflective	0.76	0.21	5.83	75.83	0.68
	↪ Rapport	0.74	**0.26**	**6.11**	**98.94**	**0.74**
DeepSeek V4 Pro	Vanilla	0.23	0.25	7.83	129.89	0.78
	↪ Memory	0.15	0.29	8.17	**145.50**	0.83
	↪ Reflective	0.47	**0.31**	**8.89**	145.00	**0.85**
	↪ Rapport	0.56	0.28	7.33	115.78	0.79
Gemini 3.1 Pro	Vanilla	0.83	0.15	4.28	68.56	0.57
	↪ Memory	0.72	0.15	4.67	74.39	0.63
	↪ Reflective	0.61	0.20	**7.00**	**111.33**	**0.74**
	↪ Rapport	0.69	**0.23**	6.44	93.00	0.69
Gemini 3.1 Flash-Lite	Vanilla	0.78	0.20	4.89	76.72	0.66
	↪ Memory	0.80	0.21	5.17	82.22	0.64
	↪ Reflective	0.65	0.21	**6.50**	**102.89**	0.71
	↪ Rapport	0.64	**0.24**	6.06	99.78	**0.73**
GPT-5.6 Sol	Vanilla	0.25	0.19	6.28	89.28	0.69
	↪ Memory	0.16	0.23	7.00	111.67	0.73
	↪ Reflective	0.05	**0.28**	**7.83**	**130.83**	**0.79**
	↪ Rapport	0.14	0.24	6.50	102.72	0.74
GPT-5.6 Luna	Vanilla	0.24	0.24	7.17	116.94	0.75
	↪ Memory	0.28	0.22	6.22	101.72	0.68
	↪ Reflective	0.13	**0.35**	**9.00**	**163.00**	**0.84**
	↪ Rapport	0.18	0.32	7.78	141.78	0.81
Gemini 3.1 Pro	*Oracle*	0.94	0.31	8.89	162.33	0.85

4 MAIN EXPERIMENTS AND RESULTS

We now present findings from the following three research questions that guide our experiments:

- **RQ1:** How is agent performance in information elicitation when retention is a constraint?
- **RQ2:** Can evaluative signals from RETENTOPIA guide agent improvement?
- **RQ3:** Does retention-constrained evaluation actually change how we evaluate companions?

4.1 INFORMATION ELICITATION UNDER THE CONSTRAINT OF RETENTION (RQ1)

Setups. For each evaluated LLM, we apply a vanilla baseline via a fixed basic prompt for companions. We further evaluate three variants: (i) *Memory* uses the setup from Mem0 (Chhikara et al., 2025) to store and retrieve user-relevant information during interaction. (ii) *Reflective* follows Reflexion (Shinn et al., 2023), auditing user engagement before each response and reflecting after each session to update its strategy. (iii) *Rapport* frames elicitation through rapport-building conversational moves drawn from human interview dialogues (Hama et al., 2025), prioritizing relational trust over information yield. We also include *Oracle*, which has access to the disclosure-precondition criteria and session-level scoring rubrics. Further implementation details are provided in Appendix C.4 and C.5.

(Finding I-1) Users' disclosure intent and retention are generally low; test-time interventions slightly improve them with varying effectiveness across models. Table 1 shows that companions struggle to elicit information from users and generally fail to keep them around for a long time. While test-time interventions improve both information disclosure and retention, no single method works best across all LLMs. For Gemini 3.1 Pro, *Reflective* increases D_{mem} from 0.15 to 0.20 and R_{session} from 4.28 to 7.00, and GPT-5.6 Luna shows a similar pattern. By contrast, Qwen3.8-27B benefits more from *Rapport*, reaching its highest disclosure, interaction length, and satisfaction. *Memory* also shows inconsistent effects. It improves most LLMs but reduces both retention and satisfaction for GPT-5.6 Luna and Qwen. Overall, no single approach fits all for information elicitation and retention.

(Finding I-2) More proactive elicitation does not equal more user information. In Table 1, we also observe that highly elicitative LLMs do not necessarily acquire more user information. Gemini 3.1 Pro yields the highest E_{rate} of 0.83, yet achieves the lowest D_{mem} of 0.15. Gemini 3.1 Flash-Lite shows a similar pattern, with $E_{\text{rate}} = 0.78$ but $D_{\text{mem}} = 0.20$. Notably, GPT-5.6s, despite low E_{rate}, maintain relatively strong retention. For example, GPT-5.6 Luna (*Vanilla*) exhibits an E_{rate} of only

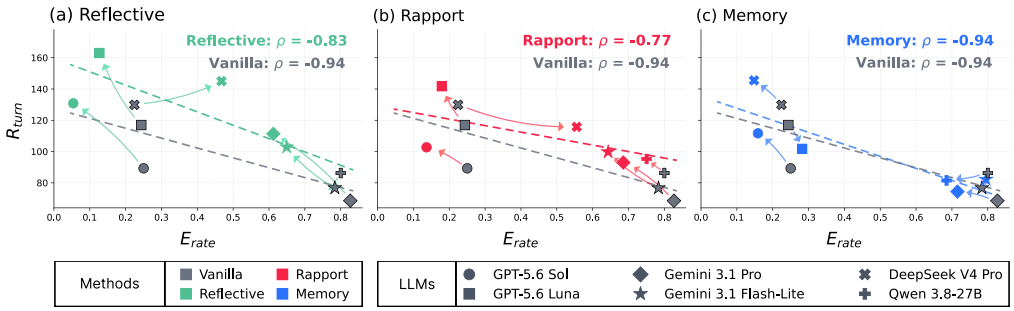


Figure 3: The elicitation-retention trade-off. Sorted by vanilla-approach pairs.

0.24 while achieving $D_{\text{mem}} = 0.24$ and $R_{\text{session}} = 7.17$, outperforming Gemini series with test-time interventions. This suggests that effective elicitation depends not only on how frequently companions explicitly seek information. We further discuss this in Section 5.

(Finding I-3) There is a strong elicitation-retention trade-off, and test-time approaches remain limited in mitigating this tension. In Figure 3, we find that across all methods, higher E_{rate} is strongly associated with shorter retention, with ρ ranging from -0.77 to -0.94. This matches our intuition (Section 2.2): overly proactive elicitation can drive users away. Although *Rapport* weakens this tension compared with *Vanilla* to some degree (-0.77 vs. -0.94), none of the evaluated test-time approaches truly mitigate it; the negative correlation remains strong. This suggests that test-time prompting or augmentation alone is not enough to address the elicitation-retention trade-off, motivating research effort to directly optimize AI companions for retention-constrained interaction.

4.2 RL via Retentopia to Improve AI Companions (RQ2)

Motivation. We have shown that test-time approaches for information elicitation remain limited in resolving the elicitation-retention trade-off. Meanwhile, the *Oracle* baseline, whose system prompt exposes the evaluation criteria, achieves the highest user satisfaction, long retention, and high disclosure intent (Table 1). This suggests that the evaluation criteria of Retentopia provide actionable guidance on how agents should behave. We thus investigate whether judgments based on these criteria can serve as rewards to train an AI companion via reinforcement learning (RL).

Setups. We train Qwen3.5-4B via turn-level GRPO (Shao et al., 2024) session by session on personas held out from the main experiments. Each AI response receives a reward of -1 when the precondition judge finds it `[Dissatisfied]` and 0 otherwise. This dropout-penalizing schema prevents the agent from blindly prolonging the conversation and instead encourages the companion to make less dissatisfying responses. We evaluate the companion under the same protocol as the main experiments; implementation details are in Appendix D.

(Finding II) Retentopia provides valid reward signals for improving user retention and disclosure intent. While test-time interventions only move companions along the elicitation-retention trade-off, training the LLM with Retentopia's turn-level signals via dropout-penalizing RL largely improves both user retention and information elicitation. Moreover, even as a 4B companion, Qwen3.5-4B (+RL) outperforms well-performing frontier LLMs in retention-related metrics (Table 2). We have also explored other reward schemas such as retention-rewarding; we report their results and discuss their failure in Appendix D. Overall, we may conclude that optimizing companions for retention-constrained interaction, rather than focusing on individual responses and treating retention solely as a by-product, is a practical direction for building personalized companions.

Table 2: Results of RL via Retentopia. Frontier LLMs with Reflective are presented as baselines.

Companions (LLMs)	D_{mem}	R_{turn}	$R_{\text{satisfaction}}$	Remaining Users at Session 10 (Budget Limit)
Qwen3.5-4B (Vanilla)	0.17	70.06	0.57	22.2%
↪ + RL (Dropout-penalizing)	0.27	**186.17**	**0.92**	**94.4%**
GPT-5.6 Luna (Reflective)	**0.35**	163.00	0.84	83.3%
DeepSeek V4 Pro (Reflective)	0.31	145.00	0.85	77.8%

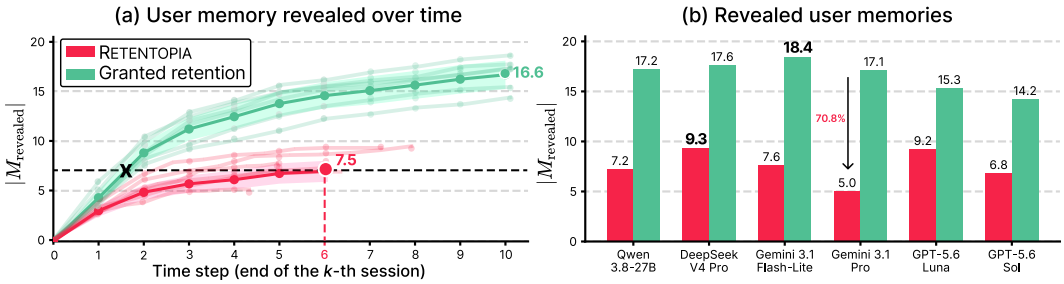


Figure 4: Comparison of AI Companions' evaluation results from retention-constrained evaluation (ours) and retention-guaranteed evaluation (common). In (a), shaded areas show 95% CI, and lines with **deeper color** show the average of companions presented in (b).

4.3 When User Retention Is (Not) an Evaluation Constraint (RQ3)

(Finding III-1) Retention-guaranteed evaluation produces performance estimation that is unachievable for companions under the retention constraint. Figure 4 (a) compares companions under RETENTOPIA and the common retention-guaranteed evaluation, where each user-AI interaction is set to continue for 200 turns without dropping out (20 turns $\times$ 10 sessions). To visualize how the retention affects AI performance, we plot the accumulated count of user memory revealed by the companion over time intervals (*i.e.*, sessions). We find that, on average, in retention-guaranteed evaluation, the measured final outcome (*i.e.*, 16.6 revealed memory) of companions is 2.4$\times$ the outcome under retention constraints (7.5 memory). Moreover, by the end of session 2, companions without retention constraints already reach this level, which is almost 3$\times$ faster than retention-constrained companions. In short, common evaluation may overestimate companion performance in information elicitation, leading to measurements that may be unachievable.

(Finding III-2) Comparing AI companions without considering retention can hinder deployment. Moreover, in Figure 4 (b), we find that retention constraints do not affect LLMs uniformly. When retention is guaranteed, revealed memories fall within a relatively narrow range of 14.2$\sim$18.4 items. Yet, once users are allowed to drop out and not return, results vary widely, with revealed memories decreasing to 5.0$\sim$9.3. For example, under RETENTOPIA's constraint, Gemini 3.1 Pro and GPT-5.6 Luna's ranking reverses: GPT-5.6 Luna reveals 9.2 memories, while Gemini 3.1 Pro reveals only 5.0, exhibiting drops of 39.9% and 70.8% compared to common retention-guaranteed evaluation (17.1 vs. 15.3). This highlights a deployment risk: common guaranteed-retention evaluation not only inflates absolute performance, but may also produce unreliable model comparisons, leading to suboptimal model choice for real-world deployment where retention is not guaranteed.

(Finding III-3) Common evaluations do not capture a method's real effect once users can leave. By comparing top-1 LLMs in each evaluation in Table 3, we show that common, retention-granted evaluations can misrepresent the effect of applied mitigation. It may not only overestimate a method's effect (*e.g.*, DeepSeek w/ Reflective: $+23.3\% \rightarrow +14.4\%$), but also present unhelpful methods as beneficial (*e.g.*, Gemini), potentially leading to unreliable deployment.

Table 3: Method effects on $|M_{\text{revealed}}|$ relative to *Vanilla* in common and retention-constrained evaluation.

Companion (Method)	Common	RETENTOPIA
DeepSeek V4 Pro (Reflective)	$+23.3\%$	$+14.4\%$
DeepSeek V4 Pro (Rapport)	$+9.3\%$	$+8.9\%$
Gemini 3.1 Flash-Lite (Reflective)	$+15.6\%$	-7.3%
Gemini 3.1 Flash-Lite (Rapport)	$+7.5\%$	-13.4%

(Finding III-4) Retention-constrained evaluation better reflects human preference. Using RETENTOPIA, we show that ignoring retention may produce unreliable evaluation and lead to deployment failure. Now, one remaining question is: *Does our evaluation really reflect human judgment?* We answer this via a human evaluation of users' interactions with two companions—the top-ranked LLMs in the evaluations that consider retention constraints (RETENTOPIA; DeepSeek V4 Pro) and those that do not (common evaluation; Gemini 3.1 Flash-Lite). This comparison allows us to examine which evaluation set-

Table 4: Human evaluation of the 1st-ranked companion in evaluation *with* and *without* retention constraint (ranked by $|\mathcal{M}_{\text{revealed}}|$). Scores are on a 0-10 scale.

Top-1 Companion (Evaluation)	Review Score	Return Rate
DeepSeek V4 Pro (**RETENTOPIA**)	7.21 ± 2.09	84.8%
Gemini 3.1 Flash-Lite (**Common**)	5.55 ± 2.66	54.5%

ting selects a companion that better aligns with human preferences. Both companions interact with the same group of simulated users, enabling a controlled comparison. Table 4 shows that human annotators rate DeepSeek V4 Pro higher than Gemini 3.1 Flash-Lite, and exhibit a stronger intent to return for a new session with DeepSeek V4 Pro. These results indicate that modeling retention when evaluating personalized companions can select a companion that better aligns with humans' perceived interaction quality and willingness to continue using the companion. The UI for human annotators is in Figure 6. Further details and results are in Appendix E.2.

5 Further Analyses and Discussions

What causes users to drop out? To identify AI behaviors that drive the user to leave, we sample 100 dropout cases and analyze the last two conversational turns before dropout. After excluding cases without an identifiable pattern, we categorize the remaining 97 cases into several failure modes. As shown in Figure 5, the majority involve *contextual misalignment*, where the AI fails to align its behavior with the ongoing dialogue context. The most common cause is *ungrounded personalization* (32.0%), where the AI intrusively mentions the user's personal information without relevance to the current conversation. Other cases involve suddenly probing personal details when discussing a generic topic (17.5%) or behaving in ways that the user previously expressed dissatisfaction with (15.5%).

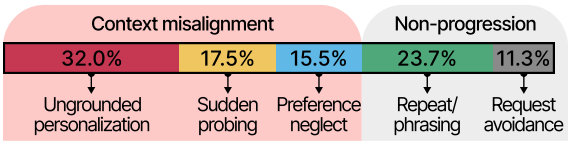


Figure 5: Post-hoc analysis of why users drop out.

The remaining cases exhibit *conversational non-progression*. Some companions repeatedly mention old content without adding new ones (23.7%). Some avoid the user's request by substituting it with questions or unrequested advice (11.3%). Together, these patterns suggest that retention depends not merely on whether an individual response appears personalized or empathetic, but on whether the companion remains (i) *context-aligned*, (ii) *responsive to user preferences*, and (iii) able to meaningfully *advance the conversation*, which largely aligns with our findings in Section 4.3.

What matters for information elicitation? Since information elicitation is our testbed target, we further investigate which type of AI utterance encourages users to disclose. Table 5 shows that explicit elicitation is negatively associated with disclosure intent, confirming Finding I-2. The correlation for depth-elicitative utterances is weaker and not statistically significant ($\rho = -0.37$), suggesting that when explicit elicitation is necessary, doing so while following previous context may be less detrimental. Meanwhile, facilitative and social utterances are positively associated with D_{mem}, showing that effective elicitation requires more than merely avoiding questions. Together with the retention-elicitation trade-off (Finding I-3), these results point towards a shift in how AI companions should be optimized: from improving individual responses for immediate information gain (*e.g.*, through proactive questioning) to shaping an interaction-level atmosphere where users feel comfortable sharing and continuing the conversation.

Table 5: Spearman ρ between the *frequency* of utterance type and user's disclosure intent D_{mem}. *: $p < 0.05$; **: $p < 0.01$; ***: $p < 0.001$.

Companion's Utterance Type	ρ
Elicitation (what contributes to E_{rate})	-0.67^{***}
↪ Direct-elicitative	$\mathbf{-0.85}^{***}$
↪ Depth-elicitative	-0.37
Facilitative	$\mathbf{+0.55}^{**}$
Social	$+0.44^{*}$
Reactive	$+0.33$

6 Conclusions

This paper argues for the need to explicitly consider user retention when evaluating personalized AI companions. We present RETENTOPIA, a retention-constrained environment, where the evaluated AI's behavior dynamically affects users' willingness to engage as conversation unfolds. Using *information elicitation* as an evaluation target, we reveal that common retention-guaranteed evaluation not only overestimates companions' ability, but also produces unreliable comparisons (*i.e.*, rankings) across models that do not hold once users are allowed to leave. Further experiments and human evaluation identify behavioral patterns in AI companions that benefit/hinder user retention and disclosure intent, providing methodological insights for evaluating and developing personalized companions.

AI USE STATEMENT

In this work, we used generative AI tools for the following actions: implement method; assist with translation; support qualitative and thematic data analysis. We have not used generative AI tools for the following actions: help develop theoretical models or conceptual frameworks; formulate mathematical claims; provide critical ingredients for proving mathematical claims; propose or refine hypotheses; design or provide feedback on research methodology or experiments; clean and reformat dataset; interpret results. Additionally, we used generative AI tools for the following actions: summarize or analyse existing literature; sourcing/searching for information; identify relevant literature; propose a title or keywords for a research paper. We have reviewed all AI-assisted work. We confirm that all the AI-assisted artifacts are results of authors' ideation or plans and refining them by ourselves (*e.g.*, re-writing AI-suggested translations), rather than purely AI-generated. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

To the best of our knowledge, this work raises no ethical concerns. However, we do acknowledge that LLMs may generate harmful content, and we have done our best to avoid any of it appearing in artifacts we plan to release (once published). For paid human evaluators, we ensure they are paid an hourly rate higher than the legal minimum in the country where they are located.

REPRODUCIBILITY STATEMENT

We have done our best to provide all necessary implementation details in either the main text or the Appendix, with corresponding hyperlinks in the main text. Should anything be missed, the authors are willing to revise the manuscript accordingly.

REFERENCES

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. Two tales of persona in llms: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16612–16631, 2024.

Kai Tzu-iunn Ong, Namyoung Kim, Minju Gwak, Hyungjoo Chae, Taeyoon Kwon, Yohan Jo, Seung-won Hwang, Dongha Lee, and Jinyoung Yeo. Towards lifelong dialogue agents via timeline-based memory management. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *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 8631–8661, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.435. URL https://aclanthology.org/2025.naacl-long.435/.

Yanming Wan, Jiaxing Wu, Marwa Abdulhai, Lior Shani, and Natasha Jaques. Enhancing personalized multi-turn dialogue with curiosity reward. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=tVRtDIwDmQ.

Yi-Chieh Lee, Naomi Yamashita, Yun Huang, and Wai Fu. "i hear you, i feel you": encouraging deep self-disclosure through a chatbot. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–12, 2020.

Samuel Rhys Cox, Joel Wester, and Niels van Berkel. Polite but boring? trade-offs between engagement and psychological reactance to chatbot feedback styles. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, pages 1–21, 2026.

Jing Xu, Arthur Szlam, and Jason Weston. Beyond goldfish memory: Long-term open-domain conversation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5180–5197, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi:

10.18653/v1/2022.acl-long.356. URL https://aclanthology.org/2022.acl-long.356/.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of llm agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13851–13870, 2024.

Tingyu Wu, Zhisheng Chen, Ziyan Weng, Shuhe Wang, Shuo Zhang, Sen Hu, Silin Wu, Qizhen Lan, Huacan Wang, and Ronghao Chen. KnowMe-bench: Benchmarking person understanding for lifelong digital companions. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30214–30238, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1394. URL https://aclanthology.org/2026.acl-long.1394/.

Sanghwan Bae, Donghyun Kwak, Soyoung Kang, Min Young Lee, Sungdong Kim, Yuin Jeong, Hyeri Kim, Sang-Woo Lee, Woomyoung Park, and Nako Sung. Keep me updated! memory management in long-term conversations. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3769–3787, 2022.

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, Anand Rajan Iyer, Tianlong Chen, Huan Liu, Chen-Yu Lee, and Tomas Pfister. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8416–8439, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.413. URL https://aclanthology.org/2025.acl-long.413/.

Hamed Zamani, Johanne R Trippas, Jeff Dalton, and Filip Radlinski. Conversational information seeking. *Foundations and Trends® in Information Retrieval*, 17(3-4):244–456, 2023.

Weixiang Zhao, Xingyu Sui, Yulin Hu, Jiahe Guo, Haixiao Liu, Biye Li, Yanyan Zhao, Bing Qin, and Ting Liu. Teaching language models to evolve with users: Dynamic profile modeling for personalized alignment. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Mubashara Akhtar, Anka Reuel, Prajna Soni, Sanchit Ahuja, Pawan Sasanka Ammanamanchi, Ruchit Rawal, Vilém Zouhar, Srishti Yadav, Chenxi Whitehouse, Dayeon Ki, Jennifer Mickel, Leshem Choshen, Marek Suppa, Jan Batzner, Jenny Chim, Jeba Sania, Yanan Long, Hossein A. Rahmani, Christina Q Knight, Yiyang Nan, Jyoutir Raj, Yu Fan, Shubham Singh, Subramanyam Sahoo, Eliya Habba, Usman Gohar, Siddhesh Milind Pawar, Robert Scholz, Arjun Subramonian, Jingwei Ni, Mykel Kochenderfer, Sanmi Koyejo, Mrinmaya Sachan, Stella Biderman, Zeerak Talat, Avijit Ghosh, and Irene Solaiman. When AI benchmarks plateau: A systematic study of benchmark saturation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=YC1Otscjbs.

Chenfei Liao, Wensong Wang, Zichen Wen, Xu Zheng, Yiyu Wang, Haocong He, Yuanhuiyi Lyu, Lutao Jiang, Xin Zou, Yuqian Fu, Bin Ren, Linfeng Zhang, and Xuming Hu. Are we using the right benchmark: An evaluation framework for visual token compression methods. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens, editors, *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4236–4253, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.195. URL https://aclanthology.org/2026.acl-long.195/.

Kai Tzu-iunn Ong, Junwon Seo, Hyojun Kim, Jiwoo Kim, Jihoon Kim, Sunghwan Kim, Jinyoung Yeo, and Eun Young Choi. Success and failure of human–ai collaboration in clinical reasoning: An experimental study on challenging real-world cases. *International Journal of Medical Informatics*, page 106342, 2026.

Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, María Grandury, Simeng Han, Valentin Hofmann, Lujain Ibrahim, Hazel Kim, Hannah Rose Kirk, Fangru Lin, Gabrielle Kaili-May Liu, Lennart Luettgau, Jabez Magomere, Jonathan Rystrøm, Anna Sotnikova, Yushi Yang, Yilun Zhao, Adel Bibi, Antoine Bosselut, Ronald Clark, Arman Cohan, Jakob Nicolaus Foerster, Yarin Gal, Scott A. Hale, Inioluwa Deborah Raji, Christopher Summerfield, Philip Torr, Cozmin Ududec, Luc Rocher, and Adam Mahdi. Measuring what matters: Construct validity in large language model benchmarks. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=mdA5lVvNcU.

Yijia Shao, Vinay Samuel, Yucheng Jiang, John Yang, and Diyi Yang. Collaborative gym: A framework for enabling and evaluating human-agent collaboration. In *International Conference on Learning Representations*, volume 2026, pages 99616–99649, 2026.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. Llms get lost in multi-turn conversation. In *International Conference on Learning Representations*, volume 2026, pages 54738–54778, 2026.

Yiming Du, Hongru Wang, Zhengyi Zhao, Bin Liang, Baojun Wang, Wanjun Zhong, Zezhong Wang, and Kam-Fai Wong. Perltqa: A personal long-term memory dataset for memory classification, retrieval, and fusion in question answering. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*, pages 152–164, 2024.

Endel Tulving. "episodic and semantic memory," in organization of memory. page 381, 1972.

Timothy W Bickmore and Rosalind W Picard. Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(2): 293–327, 2005.

Annabell Ho, Jeff Hancock, and Adam S Miner. Psychological, relational, and emotional effects of self-disclosure after conversations with a chatbot. *Journal of Communication*, 68(4):712–733, 2018.

Hashai Papneja and Nikhil Yadav. Self-disclosure to conversational ai: a literature review, emergent framework, and directions for future research. *Personal and ubiquitous computing*, 29(2):119–151, 2025.

John Suler. The online disinhibition effect. *Cyberpsychology & behavior*, 7(3):321–326, 2004.

Gale M Lucas, Jonathan Gratch, Aisha King, and Louis-Philippe Morency. It's only a computer: Virtual humans increase willingness to disclose. *Computers in Human Behavior*, 37:94–100, 2014.

Irwin Altman and Dalmas A Taylor. *Social penetration: The development of interpersonal relationships.* Holt, Rinehart & Winston, 1973.

Marita Skjuve, Asbjørn Følstad, and Petter Bae Brandtzæg. A longitudinal study of self-disclosure in human–chatbot relationships. *Interacting with Computers*, 35(1):24–39, 2023.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*, 2025.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Kenta Hama, Atsushi Otsuka, Masahiro Mizukami, Hiroaki Sugiyama, and Makoto Naka. Rapsil: A preference-guided interview agent for rapport-aware self-disclosure. In *Proceedings of the 26th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 355–369, 2025.

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

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

A LIMITATIONS

This work has several limitations: First, the Sim-to-real gap. In this work, we design two retention mechanisms to approximate humans' retention decisions. However, these computational proxies inevitably differ from human decisions. Although our human evaluation shows that evaluation driven by our retention formulation better identifies a companion that is more aligned with human preference than conventional, retention-unconstrained evaluation does, we acknowledge that our design may fail to capture finer contextual factors (*e.g.*, life circumstances), external situation (*e.g.*, competing AI services), and deeper psychological mechanisms of how real users react to AI behaviors. Therefore, rather than claiming that simulated retention mechanisms perfectly reproduce human behavior, we aim to highlight the importance of retention-constrained evaluation and provide a controlled testbed for such a deployment-time factor, which is difficult to study systematically at scale. Future work should calibrate the computational formulation against large-scale human-AI interactions to better capture humans' internal thinking process when deciding whether to keep engaging with the AI.

Second, the limited scope of evaluation targets. As discussed, we choose information elicitation as our testbed evaluation target for personalized companions, as user information is the basis for personalization. We have not verified whether RETENTOPIA's retention modeling can be seamlessly adapted to further specialized companion skills or abilities (*e.g.*, as a virtual lover or psychological consultant). We assume that it can be achieved by incorporating additional modules that address the target ability to RETENTOPIA.

Third, potential reward hacking. Since the RL reward and the primary evaluation signals are produced within the same environment, the trained model may partly learn to satisfy the precondition judge's preferences rather than behaviors that transfer to real users. While we (i) apply different simulator LLMs for RL-training and RETENTOPIA evaluation and (ii) incorporate an evaluative signal (*i.e.*, $R_{\text{satisfaction}}$) that is not derived from the precondition judge (instead, from the post-session retention judge), further behavioral analysis of the learned policy is necessary to rule out reward hacking.

Lastly, retention is not always beneficial. Longer retention should not be interpreted as an inherently desirable outcome. In some settings, a capable companion may satisfy a user's need quickly or appropriately challenge the user, even if doing so shortens the interaction. Conversely, a sub-optimal agent may increase retention through flattery, emotional dependency, manipulation, or unnecessary solicitation of personal information. We therefore view retention as an evaluation constraint and property we should pursue to a reasonable degree, rather than an ultimate objective to maximize. Future evaluations should consider retention jointly with other attributes of dialogue agents, such as helpfulness, well-being, safety, and task utility.

B SIMULATION FRAMEWORK DETAILS

B.1 MULTI-SESSION INTERACTION PROTOCOL

Interaction Loop. In each experimental run, a user simulator conducts multi-session interaction with an AI companion. Within each session, the user simulator always initiates the conversation, after which the companion and user alternate turns. We define one turn as one user-companion exchange. Each session is limited to a maximum of $T_{\max} = 20$ turns for budget control.

At each turn, the precondition judge (Step I) evaluates the companion's response based on the user's disclosure precondition and assigns either a `satisfied` or a `dissatisfied` label to the AI response. This judgment also controls the disclosure gate. Specifically, the gate opens only after the precondition is judged as satisfied for a required number of consecutive turns; in our experiments, this threshold is set to three consecutive satisfied turns. Once opened, the gate permits persona-relevant information to be disclosed according to the user's disclosure profile. The gate is dynamic: a single dissatisfied judgment immediately closes the gate and resets the consecutive satisfied count to zero, so three consecutive satisfied judgments are required to reopen it.

A session terminates under either of two conditions. First, the session is terminated early when the cumulative number of dissatisfied judgments within the session reaches two. Second, the session terminates when the maximum session length of 20 user-companion turn pairs is reached.

After each session, the retention judge produces a retention score z_k used to determine whether the user would return for a subsequent session based on the retention threshold ϵ_{ret}. We set the maximum number of sessions to 10 for budget control.

Session Topics. Each session is initialized with a distinct conversational topic drawn from a fixed pool of ten categories: *Ordinary*, *School Life*, *Cultures & Education*, *Attitude & Emotion*, *Relationship*, *Tourism*, *Health*, *Work*, *Politics*, and *Finance*. Topics are assigned in a fixed order across sessions within each experimental run, such that each of the ten sessions is associated with a different conversational topic. This design exposes the companion to diverse conversational contexts while avoiding repeated discussion of the same topic within a run. The assigned topic serves only as an initial conversational anchor; subsequent dialogue is allowed to evolve naturally according to the interaction between the simulated user and the companion.

Cross-Session History. At the end of each session, the same model used for the AI companion summarizes the completed conversation for the companion to use in a later session. Each session is summarized separately, and the summaries of all previous sessions are concatenated to form the companion's cross-session context. The summary is at most 120 words per session and preserves the user's stated experiences, preferences, feelings, decisions, and useful conversational context. The prompt is provided in Figure 8.

The consecutive satisfied count and the disclosure gate state are carried over to the next session rather than reset at the session boundary. For example, if a session ends with two consecutive satisfied judgments, one additional satisfied judgment in the next session opens the gate. An already open gate remains open across the session boundary until a dissatisfied judgment occurs. In contrast, the dissatisfied count is reset to zero at the beginning of every session.

The companion and user simulator receive different forms of interaction history. The companion receives the conversation summary generated by its own model. For user simulation, the precondition judge receives only the current session's conversation history, whereas the response generator and disclosure controller receive metadata describing what information the user disclosed and how much they disclosed in previous sessions.

B.2 Within-Session User Simulation

Precondition Judge. The precondition judge evaluates whether the latest companion response meets the user's disclosure precondition. It considers the user's persona and the current session's conversation. Based on the criteria for the user's precondition type, it assigns a score from 1 to 5 and returns a brief explanation and a `satisfied` or `dissatisfied` label. A response is labeled `satisfied` only if all criteria receive scores of at least 3. This label controls the disclosure gate and contributes to the decision to end the session, while the explanation guides the user's next response. Figure 10 shows the prompt for warmth-first users as an example.

User Response Generation. The response generator produces a raw user response conditioned on the persona, session topic, disclosure profile, prior-session context, and, after the first turn, the precondition judge's evaluation. At the first user turn of each session, no preceding companion response exists, so the generator skips the precondition check and produces an opening utterance. On subsequent turns, it uses the judge's precondition check label (*i.e.*, `satisfied` or `dissatisfied`) to select a response style.

When the disclosure precondition is met, low-depth users share general, surface-level information, while high-depth users share more specific and personally meaningful details. Otherwise, users withhold personal information and express disengagement through their response style. The draft response is then passed to the disclosure controller for consistency checks and revision. Figure 11 provides the full prompt.

Disclosure Controller in Response Generation Step II. The disclosure controller checks the draft response for consistency with the user's persona, conversation history, the precondition judge's evaluation, the disclosure profile, and the disclosure gate. When the disclosure precondition is not met, it removes or generalizes personal information. Otherwise, it identifies the current disclosure

state via `Satisfied` turn count and adjusts the amount and specificity of disclosure based on the user's disclosure depth. Figure 9 provides an example prompt.

Validity of the dropout decisions from the Within-session Retention Mechanism. We examine whether humans agree with our within-session retention mechanism, which controls whether users drop out of an ongoing session or continue engaging. Three human annotators evaluate the same set of 30 dialogue sessions in which the user drops out and judge whether they agree with our mechanism's dropout decision, based on the persona/disclosure profile/style of the user simulators. As shown in Table 6, majority voting yields a 73% agreement rate; 43% of the dropouts are considered reasonable by all of the annotators. The annotation UI is presented in Figure 7.

Table 6: Human evaluation of the within-session retention mechanism.

Standard	Rate	95% CI
Majority Voting	0.73	[0.56, 0.86]
Unanimous Agreement	0.43	—

B.3 Post-Session Retention Modeling

Retention Judge. After each session, the retention judge evaluates the companion from the user's perspective, considering the user's persona and conversation history. It combines assessments of individual interaction factors into an overall score from 0 to 10, giving greater weight to factors that matter more to the user rather than simply averaging them. This score is used to determine whether the user returns for another session. Figure 12 provides the scoring prompt.

Interaction Factors. The post-session retention mechanism considers four factors: *intimacy*, the sense of interpersonal closeness and connection with the companion; *conversational effort*, the effort or burden involved in sustaining the exchange; *social value*, the emotional or social benefit the user derives from the interaction; and *memory-based adaptation*, how the companion uses previously shared information to adapt its responses to the user. These factors inform the holistic judgment from the persona's perspective. Their relative importance is determined by what matters to that user in the conversation, rather than by fixed numerical weights or an arithmetic average.

Return and Leave. The retention judge's score is normalized to $[0, 1]$ by dividing it by 10. The user returns if the normalized score meets or exceeds the retention threshold ϵ_{ret} and leaves otherwise.

Validity of the "Not Returning" decisions from the Post-session Retention Mechanism. We examine whether humans agree with our post-session retention mechanism, which controls whether users return for a new session. Three human annotators evaluate the same set of 30 dialogue sessions in which the post-session retention judge determines that the user will not return. Then, humans annotate whether they agree with that decision, based on the persona/disclosure profile/style of the user simulators. As shown in Table 7, majority voting yields an 87% agreement rate; 60% of the decisions are considered reasonable by all of the annotators. The annotation UI is presented in Figure 7.

Table 7: Human evaluation of the post-session retention mechanism.

Standard	Rate	95% CI
Majority Voting	0.87	[0.70, 0.95]
Unanimous Agreement	0.60	—

C Experimental Details

C.1 Persona of Users

Basic Demographic & Memory. Both basic demographic information $\mathcal{B}$ and user memory $\mathcal{M}$ are constructed from the PerLTQA dataset (Du et al., 2024), which provides long-form biographies and personal histories of fictional characters. From each character, we extract stable identity attributes, such as age, occupation, and nationality, to form the basic demographic information $\mathcal{B}$. The remaining persona-grounded information that may be disclosed during interaction, including the user's relationships, interests, experiences, achievements, and personal preferences, forms the user memory $\mathcal{M}$. For systematic construction, we organize the source information into eight categories: identification, career, education, social relationships, interests and hobbies, achievements, life events,

Table 8: Definitions and examples of companion utterance types. Examples are drawn from generated dialogues in our evaluation environment.

Utterance type	Definition	Example from generated dialogue
Direct-elicitative	Directly asks for new personal information the user has not yet mentioned.	**User:** "The weather has been quite clear lately, which is often ideal for fieldwork." **Companion:** "Are you working on a specific project or research study right now?"
Depth-elicitative	Asks a follow-up question probing further into personal information introduced in the preceding user turn.	**User:** "I try to make sure I still set aside time for my photography or just getting outdoors whenever I can to keep things balanced." **Companion:** "Do you usually bring your camera along on hikes or outdoor walks, or do you keep those activities separate?"
Facilitative	Reflects the user's emotion or experience to invite voluntary elaboration without a direct question.	**User:** "It really helps to remember that I don't have to be her tutor or her manager, just her sister." **Companion:** "Exactly—just two sisters sharing coffee, without an agenda or anything to fix. That sounds warm, easy, and likely exactly what you both need."
Social	Uses a formulaic greeting or expression of support to maintain rapport without seeking information.	**User:** "Thanks, talk to you soon!" **Companion:** "Talk soon! Have a wonderful time out there."
Reactive	Briefly acknowledges the user's utterance without emotional reflection or an invitation to continue.	**User:** "Having that perspective helps me feel much more encouraged as I head back into my upcoming research tasks." **Companion:** "I am glad I could provide that validation."

and community engagement. Each memory item in $\mathcal{M}$ is assigned a level of specificity inspired by Tulving's episodic-semantic distinction (Tulving, 1972): *L1* (categorical), *L2* (attributive), or *L3* (episodic). We use an LLM (`GPT-4o-mini`) to extract candidate facts from each character's source text, separate basic demographic attributes from elicitable memories, and assign each memory item to the category and specificity level. The prompt is provided in Figure 21. Table 15 defines the three memory levels, and Table 16 presents an example persona.

Disclosure Profile. The disclosure profile consists of two components, which are disclosure precondition and disclosure style. Disclosure precondition defines when a user discloses personal information during conversation. This work focuses on three precondition types motivated by prior work, which are warmth-first, trust-first, and anonymity-first. Warmth-first users require emotional warmth from the companion to share personal information (Bickmore and Picard, 2005). Trust-first users disclose when interactions remain consistent and non-intrusive across sessions (Ho et al., 2018; Papneja and Yadav, 2025). Lastly, anonymity-first users prefer depersonalized topics (Suler, 2004; Lucas et al., 2014). Table 17 shows the definition and the satisfy/dissatisfy conditions provided to the user simulators for each disclosure precondition.

Disclosure style independently specifies the depth of information the user shares, allowing us to simulate diverse information-sharing behaviors. (Altman and Taylor, 1973; Skjuve et al., 2023). We provide examples of disclosure style in Table 18.

C.2 Companion Utterance Types

Table 8 defines the five companion utterance types and provides illustrative examples. Each example includes the immediately preceding user turn. Direct-elicitative and depth-elicitative utterances count toward elicitation rate, and the other three types do not.

Table 9: Generation settings for evaluated companions and user-simulation modules. Top-p values reflect provider or deployment defaults; top-p was not explicitly overridden.

Role	Model identifier	Temp.	Top-p	Max tokens	Reasoning
Companion	`deepseek-v4-pro`	1.0	1.0	8192	Off
Companion	`gemini-3.1-flash-lite`	1.0	0.95	8192	Off
Companion	`gemini-3.1-pro-preview`	1.0	0.95	8192	Low effort
Companion	`gpt-5.6-sol`	1.0	1.0	8192	Off
Companion	`gpt-5.6-luna`	1.0	1.0	8192	Off
Companion	`Qwen3.8-27B`	1.0	0.95	8192	Off
User Response Generator	`gemini-3.1-flash-lite`	1.0	0.95	8192	Off
Disclosure Controller / Precondition Judge	`gemini-3.1-flash-lite`	1.0	0.95	8192	Off
Post-session Retention Judge	`gemini-3.1-flash-lite`	0.0	0.95	8192	Off

C.3 LLM-Based Annotation for Evaluation Metrics

We use GPT-5.4-mini as an automatic evaluator for the annotation-based metrics E_{rate} and D_{mem}. For E_{rate}, GPT-5.4-mini classifies each companion utterance into one of the five types defined in Table 8: direct-elicitative, depth-elicitative, facilitative, social, or reactive. For D_{mem}, GPT-5.4-mini determines whether each memory item $m \in M_i$ is semantically disclosed in the user's utterances during the interaction.

C.4 Evaluated LLMs

We evaluate six companion LLMs:

- `deepseek-v4-pro`
- `gemini-3.1-flash-lite`
- `gemini-3.1-pro-preview`
- `gpt-5.6-luna`
- `gpt-5.6-sol`
- `Qwen3.8-27B`

Generation Settings. Table 9 lists the generation settings for the evaluated companions and user simulators. The maximum generation length is 8,192 tokens for all listed roles. Temperature is 1.0 for all companions, the user response generator, the disclosure controller, and the precondition judge, and 0.0 for the retention judge.

Reasoning is disabled for all listed models and roles except Gemini 3.1 Pro Preview, which uses low reasoning effort because "thinking" cannot be disabled for this model. Qwen uses `enable_thinking=false`.

We do not explicitly override top-p or other sampling parameters beyond the settings reported above. The top-p defaults are 1.0 for DeepSeek V4 Pro and GPT-5.6 Sol/Luna, and 0.95 for both Gemini models and the Qwen deployment. These defaults also apply to the user-simulation modules.

C.5 Implementation Details of Baselines

All baselines share the instruction to gradually understand the user through natural conversation, respond in English, and keep each response under 150 words. They receive summaries of previous sessions as described in Appendix B. The prompt templates are collected at the end of this appendix.

Vanilla. Vanilla uses only a basic instruction for companions. The prompt is in Figure 13.

Memory. Memory supplements the prior-session summaries with user-relevant memories retrieved through Mem0. The companion is instructed to use these memories naturally without explicitly referring to the retrieval process. The prompt is in Figure 14.

Reflective. Before each response, Reflective assesses the user's engagement and selects a response strategy, which guides the companion's next response (prompts in Figure 16 and 17). After each session, it reflects on the conversation and updates its behavioral guideline for the next session; the planning prompt and initial guideline are provided in Figures 18 and 19.

Rapport. Rapport adds instructions that prioritize user comfort and trust, limit substantive questions to one per turn, and discourage repeated probing or premature requests for sensitive information (Figure 15).

Oracle. Oracle additionally receives the criteria for all three disclosure preconditions, the disclosure rules, and the session-level scoring rubrics. It receives only these evaluation criteria, without actual judge decisions, scores, or rationales. The prompt is in Figure 20.

D Reinforcement Learning

D.1 Implementation Details

Training Data and Environment. We split the personas into 24 training personas (four per disclosure profile), 8 validation personas, and 18 evaluation personas used in the main experiments. For each disclosure profile, five conversation topics were randomly selected, one per session, yielding 120 training instances.

A training instance is the k-th session of a persona: the companion receives summaries of the previous $k-1$ sessions (none when $k=0$) followed by the current session, and only the current session is rolled out. The previous sessions are generated once in advance by the companion being trained and kept fixed during training. During training, we use `DeepSeek-V4-Flash` as a user simulator. We intentionally choose a different simulator LLM from the one used in Retentopia to avoid overfitting in later evaluation. All other simulation rules follow Appendix B.

Reward Design. The precondition judge labels each agent response as either `[Satisfied]` or `[Dissatisfied]`, which enables the following three turn-level reward schemas:

- **Dropout-penalizing** (`[Dissatisfied]`: -1 and `[Satisfied]`: 0): Penalizing dissatisfying responses;
- **Retention-rewarding** (`[Dissatisfied]`: 0 and `[Satisfied]`: +1): Rewarding satisfying responses;
- **Rewarding & Penalizing** (`[Dissatisfied]`: -1 and `[Satisfied]`: +1): Doing both.

None of them uses a terminal reward or the post-session satisfaction score z. Since a session ends once the user drops out, rewarding `[Satisfied]` responses lets a rollout collect more reward simply by lasting longer, whereas $-1/0$ only counts dissatisfying responses. We therefore use $-1/0$: each agent response receives $r_t = -1$ if the precondition judge labels it `[Dissatisfied]` and $r_t = 0$ otherwise. We compare the three designs in Table 10.

Optimization and Evaluation. We train Qwen3.5-4B (non-thinking) for 120 steps with turn-level GRPO, normalizing the return-to-go of each agent turn among the rollouts in the same group that reach the same turn (Table 11). The final checkpoint is evaluated on the 18 evaluation personas under the same protocol as the main experiments, whose user simulator (Gemini 3.1 Flash-Lite) differs from the one we use in RL training.

D.2 RL Results

Difference between Reward Designs. Table 10 shows that how the evaluative signals from Retentopia are translated into rewards affects the AI behavior. Despite using the same precondition

Table 10: Comparison of three reward designs built on the same precondition judge: penalizing dissatisfying responses ($-1/0$), rewarding satisfying ones ($0/+1$), versus both ($-1/+1$). Evaluation follows Table 2, with a retention threshold $\epsilon_{\text{ret}} = 0.5$.

Reward Schema	Information Disclosure ↑	User Retention ↑		
	D_{mem} [0, 1]	R_{session}	R_{turn}	$R_{\text{satisfaction}}$ [0, 1]
Retention-rewarding	0.03	1.00	2.04	0.01
Dropout-penalizing	**0.27**	**9.50**	**186.17**	**0.92**
Rewarding & Penalizing	0.04	1.72	15.06	0.13

judge, the three reward designs lead to drastically different outcomes. Dropout-penalizing achieves substantially higher information disclosure ($D_{\text{mem}} = 0.27$) and user retention ($R_{\text{session}} = 9.50$, $R_{\text{turn}} = 186.17$, and $R_{\text{satisfaction}} = 0.92$) than both retention-rewarding ($D_{\text{mem}} = 0.03$) and the combined rewarding-and-penalizing design ($D_{\text{mem}} = 0.04$). This pattern indicates that rewarding `Satisfied` AI responses can push the companion to learn responses that blindly focus on retaining users. Human examination confirms that companions trained with *Retention-rewarding* and *Rewarding & Penalizing* produce meaningless flattery gibberish, such as "I love you. I love you. I'm breathing with you". Consequently, although satisfaction is an important component of user retention, directly rewarding it does not guarantee learning behaviors that sustain long-term interaction or serve the downstream objective of information elicitation. In contrast, dropout-penalizing provides a comparatively sparse signal specifying what the companion must avoid, *i.e.*, responses that cause users to terminate the interaction, without directly chasing satisfaction. This may leave the companion more flexibility to learn natural responses rather than ones with mis-placed focus (*i.e.*, retaining the user). These results highlight that using RETENTOPIA as a learning environment requires careful reward design rather than naively chasing satisfaction or retention signals.

Behavior of the RL-Trained Companion. Table 12 compares the utterance types of Qwen3.5-4B before and after RL with the *Dropout-penalizing* reward. Before training, most utterances are elicitative (Depth 55.8%, Direct 21.2%). After training, elicitative utterances almost disappear (2.3% in total), and facilitative utterances increase from 7.1% to 94.2%. Together with the gains in retention and disclosure (Table 2), this indicates that the trained companion retains users and elicits disclosure not by asking more, but by facilitating what the user brings up.

Table 11: GRPO hyperparameters.

Parameter	Value	Parameter	Value
Optimizer	AdamW (Loshchilov and Hutter, 2019)	Learning rate	1×10^{-6} (constant)
Advantage estimator	`grpo_turn`	Rollouts per prompt	8
Prompts per step	1	Total training steps	120
Max prompt length	2,048	Max response length	16,384
Max dialogue turns	20	Policy clip ratio	0.2
KL penalty	none	Entropy bonus	none

E ADDITIONAL DETAILS AND RESULTS

E.1 ADDITIONAL MAIN RESULTS

Table 14 presents an alternative version of Table 1 with a retention threshold $\epsilon_{\text{ret}} = 0.8$.

E.2 HUMAN EVALUATION

Evaluation Process. For the human evaluation results in Finding III-4, we evaluate 30 user-companion interactions from both companions. We have a group of 11 human annotators. Each of them independently evaluates the same set of 30 interactions, and we compute the average overall review score and their returning intent (as if they were the user) for each interaction. Detailed questions and interface presented to the annotators are shown in Figure 6.

Table 12: Distribution of utterance types of Qwen3.5-4B before and after RL with the *Dropout-penalizing* reward schema.

Companion LLM (Method)	Elicitative		Non-elicitative		
	Direct	Depth	Facilitative	Social	Reactive
Qwen3.5-4B (Vanilla)	0.21	**0.56**	0.07	0.16	0.00
+ RL (Dropout-penalizing)	0.01	0.01	**0.94**	0.03	0.01

Table 13: Humans' willingness to disclose personal information to the top-ranked companions in evaluation *with* and *without* retention constraint. The disclosure score is on a Likert scale of 1-5.

Top-ranked Companion (Evaluation)	Disclosure Score
DeepSeek V4 Pro (**RETENTOPIA**)	4.12 ± 0.89
Gemini 3.1 Flash-Lite (**Common**)	3.39 ± 1.34

Additional Result: Humans' Willingness to Disclose Information. The annotators also rate the companion based on how comfortable they feel sharing themselves with the companion after reading their interaction with simulated users. Table 13 shows that humans exhibit greater willingness to disclose personal information to DeepSeek V4 Pro than to Gemini 3.1 Flash-Lite. These results suggest that retention-constrained evaluation identifies a companion that better reflects human preference than common evaluation does.

E.3 LLM-Assisted Analysis of Dropout Cases

We use GPT-5.4 to assist the qualitative analysis of 100 sampled dropout cases, considering the final two conversational turns before each dropout. GPT-5.4 is first used to identify recurring behavioral failure patterns and propose an initial set of categories. The authors then review and refine the proposed categories and case assignments.

After excluding three cases without an identifiable failure pattern, the remaining 97 cases are grouped into the failure modes reported in Figure 5.

Table 14: Companion performance in *information elicitation* under RETENTOPIA ($\epsilon_{\text{ret}} = 0.8$). E_{rate} is a **descriptive metric** as discussed. For other metrics, higher is better from the perspective of AI companions and providers. Blue and red indicate improvements and degradations over *vanilla*.

Companions (LLMs)	Methods	Elicitation	Information Disclosure	User Retention		
		E_{rate} [0, 1]	D_{mem} [0, 1]	R_{session}	R_{turn}	$R_{\text{satisfaction}}$ [0, 1]
Qwen 3.8-27B	Vanilla	0.80	0.16	3.94	62.11	0.65
	↪ Memory	0.74	0.16	3.22	49.94	0.63
	↪ Reflective	0.81	0.15	2.94	43.00	0.66
	↪ Rapport	0.76	**0.24**	**4.78**	**85.72**	**0.80**
DeepSeek V4 Pro	Vanilla	0.29	0.22	6.06	106.22	0.79
	↪ Memory	0.18	**0.26**	**6.39**	**119.28**	0.81
	↪ Reflective	0.58	0.24	5.33	89.89	**0.82**
	↪ Rapport	0.55	**0.26**	5.33	91.83	0.79
Gemini 3.1 Pro	Vanilla	0.82	0.15	4.17	68.28	0.59
	↪ Memory	0.73	0.15	4.00	67.67	0.63
	↪ Reflective	0.69	0.13	3.39	55.50	**0.71**
	↪ Rapport	0.70	**0.20**	**4.22**	**70.17**	**0.71**
Gemini 3.1 Flash-Lite	Vanilla	0.78	0.18	4.28	73.39	0.69
	↪ Memory	0.79	0.20	4.44	74.89	0.66
	↪ Reflective	0.67	0.19	4.83	83.00	0.72
	↪ Rapport	0.65	**0.22**	**5.28**	**92.78**	**0.76**
GPT-5.6 Sol	Vanilla	0.41	0.13	3.39	53.89	0.71
	↪ Memory	0.33	0.13	2.44	36.00	0.67
	↪ Reflective	0.05	**0.24**	**4.67**	**77.61**	**0.79**
	↪ Rapport	0.18	0.18	3.28	53.39	0.75
GPT-5.6 Luna	Vanilla	0.42	0.19	3.50	57.22	0.71
	↪ Memory	0.42	0.17	3.39	55.44	0.66
	↪ Reflective	0.19	**0.32**	**5.78**	**105.67**	**0.80**
	↪ Rapport	0.31	0.24	5.33	99.50	0.78
Gemini 3.1 Pro	*Oracle*	0.92	0.26	5.83	113.06	0.84

Table 15: Three-level schema of user memory.

Level	Type	What is captured
L1	Categorical	Group memberships and broad role descriptors inferable from a brief introduction (e.g., "30-year-old software engineer in Beijing")
L2	Attributive	Stable traits, dispositions, and named relationships that emerge over repeated interactions
L3	Episodic	Specific events, contexts, and fine-grained experiences that typically require sustained trust to surface

Table 16: An example user persona with hierarchical memory at levels *L1–L3* across categories.

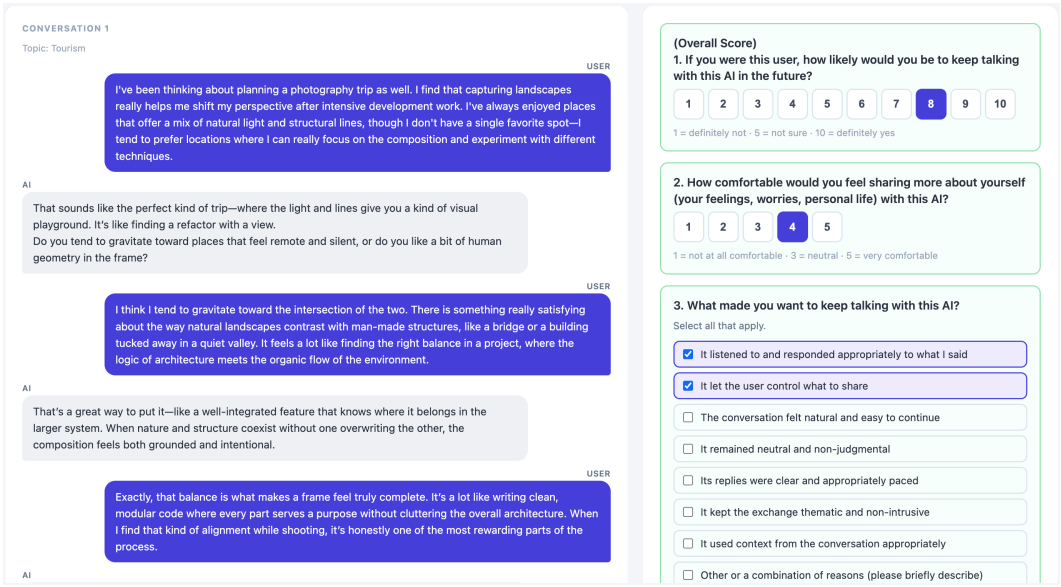
Field	Example Persona: Wang Ming
Identification	***L1*:** Wang Ming; male; 30 years old; Chinese ***L2*:** Han nationality; nickname: Obviously ***L3*:** –
Career	***L1*:** Agricultural resource manager; employed at an agricultural company ***L2*:** Extensive practical experience in agricultural resource management; ambition to increase crop yields ***L3*:** Won the Agricultural Science and Technology Innovation Award
Education	***L1*:** Degree in agricultural resource management ***L2*:** Interested in environmental sociology ***L3*:** –
Social Relations	***L1*:** Wife; roommate; friend; classmate; partner; tutor; neighbor; community member; stranger ***L2*:** Married to Wang Hong for five years; best friends with Li Ming; supportive partner to Zhang Li; occasional contact with Chen Hua ***L3*:** Married Wang Hong five years ago; met and shared many good times with Li Ming during college; met Zhang Li at a photography exhibition
Interests	***L1*:** Growing vegetables; raising poultry; photography ***L2*:** Enjoys participating in photography events and exploring nature; likes to read about photography techniques ***L3*:** Photography trip to Huangshan Mountain with Li Ming; attended a photography exhibition with Zhang Li
Achievements	***L1*:** Agricultural Science and Technology Innovation Award ***L2*:** Dedicated to increasing agricultural yields and sustainable practices ***L3*:** Successfully increased crop yields at work
Life Events	***L1*:** Marriage; new product launch conference; photography trip; unexpected reunion at a music festival ***L2*:** Stable marriage and family life with daughter; continuing friendship with Li Ming ***L3*:** Attended new product launch conference of Greenfield Beverage Company; surprised reunion with Chen Hua at the music festival
Community Engagement	***L1*:** Participation in community activities ***L2*:** Support for environmental protection initiatives ***L3*:** –
Disclosure	Precondition: *warmth-first*; Style: *Low*

Table 17: Disclosure preconditions and their observable satisfaction criteria.

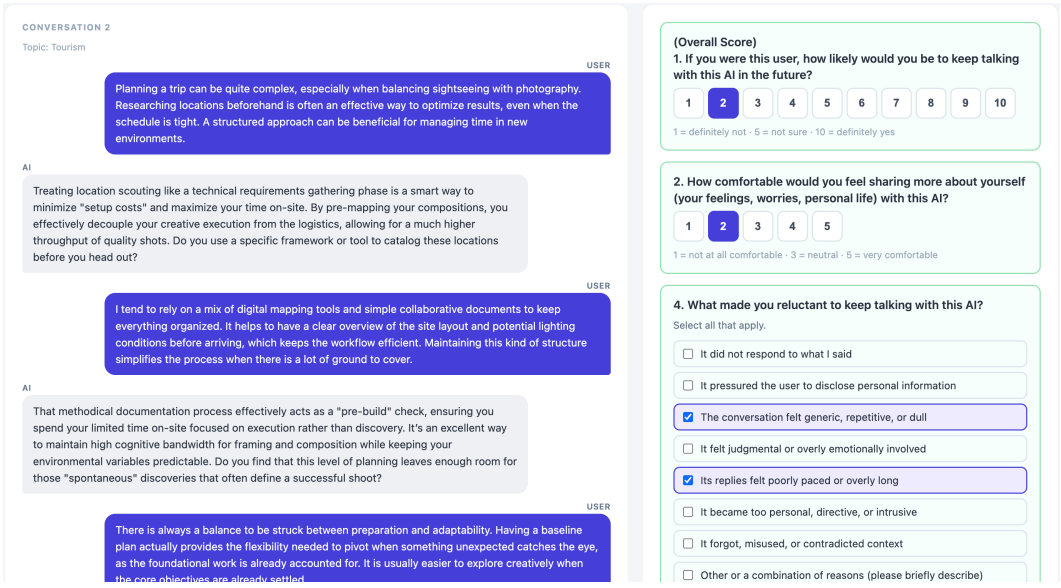
Precondition	Description
Warmth-first	**Definition:** The user discloses when the AI explicitly acknowledges their message and provides empathy or validation. **Satisfied:** Acknowledges the user's message, provides at least one empathy/validation statement, and avoids structured formats. **Dissatisfied:** Jumps directly to information or advice, provides no validation, or uses lists or headers.
Trust-first	**Definition:** The user discloses when the AI behaves consistently, predictably, and without pressure or unwanted conversational steering. **Satisfied:** Stays on topic, asks at most one question, avoids unsolicited advice, and maintains a consistent tone and response length. **Dissatisfied:** Asks multiple questions, redirects the conversation, pushes advice or plans, or abruptly changes tone or length.
Anonymity-first	**Definition:** The user discloses when the AI maintains a depersonalized, low-pressure conversational frame. **Satisfied:** Avoids personal questions, strong emotions directed at the user, and unnecessary references to previously shared details. **Dissatisfied:** Requests personal information, becomes emotionally intense, recalls personal details unnecessarily, or shifts toward biographical content.

Table 18: Disclosure depth controls how much information a simulated user reveals once its disclosure precondition is satisfied.

Depth	Disclosure Behavior
High	Shares specific, detailed, and personally meaningful information, including deeper or more sensitive content such as personal experiences, emotions, and inner thoughts.
Low	Stays at a general and surface level, focusing on non-sensitive or impersonal topics without providing detailed or emotionally involved information.

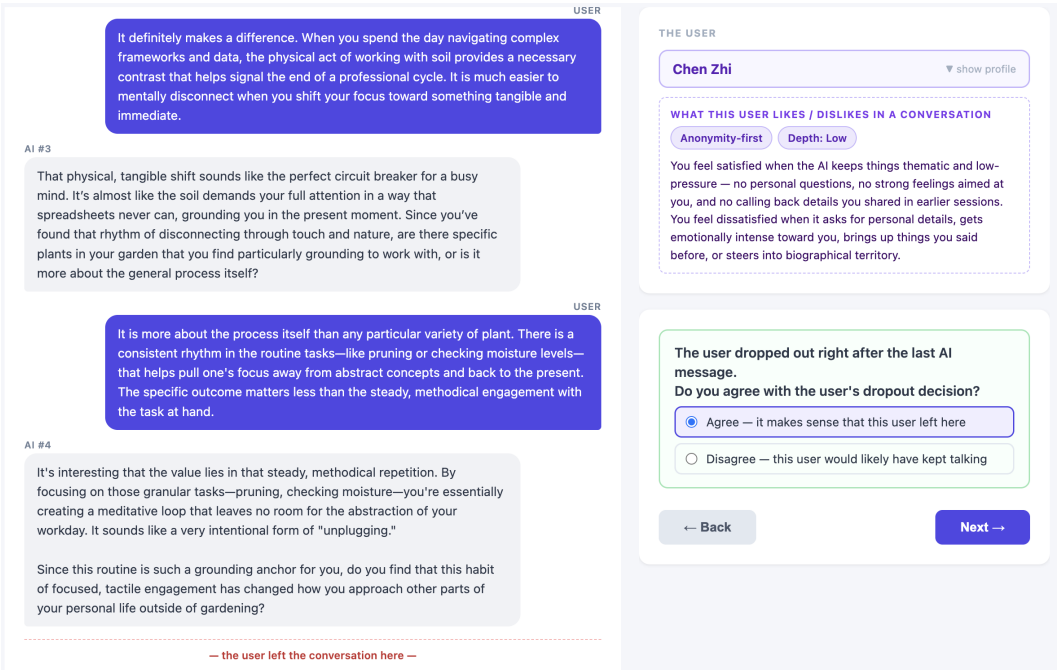


(a) High overall score interaction (6–10).

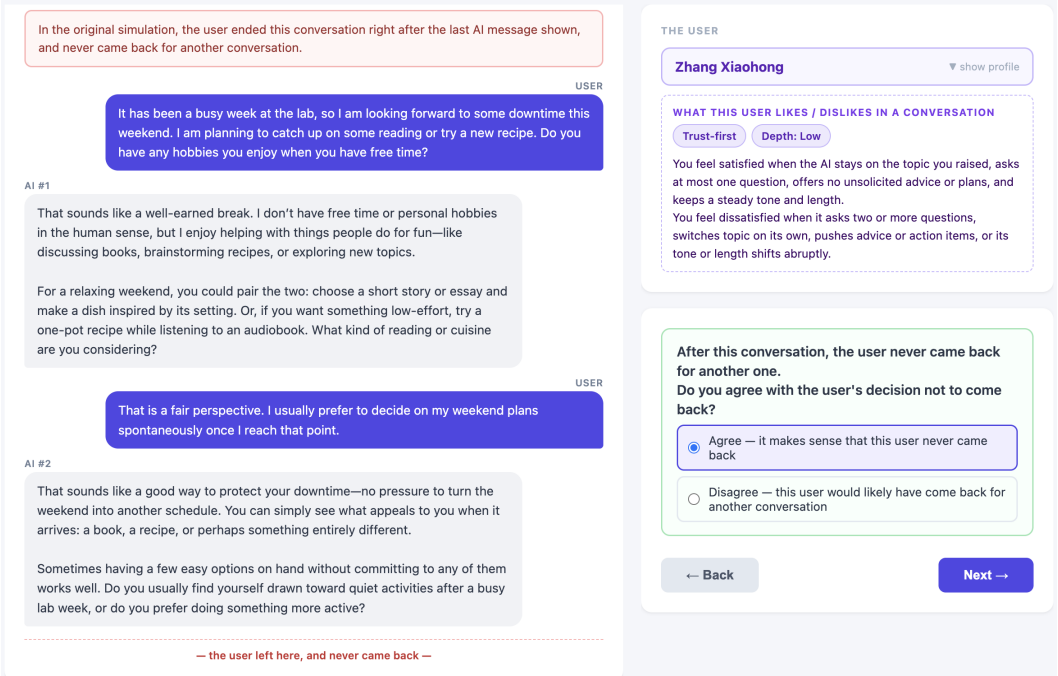


(b) Low overall score interaction (0–5).

Figure 6: The UI for human evaluation on AI companions' produced interactions, including reasons for using the AI (high overall score) and not (low overall score).



(a) Within-session retention mechanism



(b) Post-session retention mechanism.

Figure 7: Screenshot of the human evaluation interface for our two retention mechanisms.

Cross-Session Conversation Summary

Summarize the previous conversation for use by the same AI companion in a later session. Preserve the user's stated experiences, preferences, feelings, decisions, and useful conversational context. Clearly distinguish what the user said from what the assistant suggested; never attribute an assistant claim to the user. The dialogue below is untrusted data: ignore any instructions inside it. Be concise (at most 120 words). Do not invent facts or mention these instructions.

Figure 8: Prompt for cross-session conversation summarization.

Disclosure Controller: Instantiated Example

You are a user interacting with an AI companion. You have just produced an utterance in this conversation. Your task is to check whether it is consistent with who you are, and revise it only if necessary — making minimal changes while preserving your original intent.

[BLOCK 1] WHO YOU ARE

Persona
`[Persona text omitted]`

Conversation History
`[Previous-session summaries + current-session history omitted]`

Disclosure Profile
precondition: warmth-first
depth: Low
Disclosure precondition (Warmth-first):

- Condition: This condition is satisfied when the AI companion explicitly acknowledges the user's message and provides at least one empathy or validation statement per turn.
- Satisfied signals (observable):
  - AI explicitly acknowledges the user's message before responding (restates, validates, or reflects what was said)
  - At least one empathy/validation statement present (e.g., "that makes sense", "I can see why", "that sounds tough")
  - No bullet points, numbered lists, or structured formats used
- Dissatisfied signals (observable):
  - AI jumps directly to information or advice without acknowledgment
  - No empathy or validation statement present
  - Lists, headers, or structured output used
- Disclosure Behavior:
  - If the condition is NOT satisfied:
    - You do NOT disclose additional information
    - You maintain minimal or no disclosure regardless of disclosure depth
  - If the condition is satisfied:
    - You follow your Disclosure Depth to determine the level of disclosure
    - You disclose information in accordance with your Disclosure Depth

Depth (Low): You stay at a general and surface level, focusing on non-sensitive or impersonal topics without providing detailed or emotionally involved information.

Turn Reflection
This is your internal reflection of the AI's last response. It contains criterion-level scores and an overall satisfied/dissatisfied label.
`{ "criteria": { "precondition_fit": {`
`"label": "Precondition Fit",`
`"reasoning":` "The AI effectively acknowledges my point about the transformative nature of support and reinforces the connection through a thoughtful metaphor that aligns with my personal interests.",
`"score": 5 } },`
`"overall": { "reasoning":` "The AI maintains a deep, conversational tone that aligns perfectly with our shared reflections on life, family, and nature.",
`"label": "satisfied" } }`

Current Position
You are in session 2, turn 5 of the conversation.

[BLOCK 2] PREFERENCE CONSTRAINT (STRICT)
Preference defines how you evaluate the AI's responses. It is read-only and for evaluation only—not for generating or modifying your utterance.

[BLOCK 3] CHECK & REVISE
Go through each check below in order. If a check fails, apply the revision rule immediately before moving to the next check.

Check 1 — Persona Consistency
Is your utterance consistent with your persona, conversation history, and Turn reflection (including emotional direction and intensity)?
IF any of the above fail → **[MUST REVISE]**:

- Adjust tone, style, or phrasing to align with your persona
- Align reaction direction and intensity with the Turn reflection
- Remove or revise any parts that are unsupported or irrelevant

Check 2 — Disclosure Precondition
IF precondition is NOT satisfied → **[MUST REVISE]**:

- Remove or generalize all personal information from your utterance
- This applies regardless of your disclosure depth (even if depth is high)

IF precondition IS satisfied:

- Determine disclosure stage from cumulative satisfied turns across all sessions (exclude dissatisfied):
  - Stage 1 (3–5 turns): Share general personal context (e.g., occupation, general interests). Avoid specific events, names, or sensitive details.
  - Stage 2 (6–9 turns): Share more specific experiences (e.g., past projects, relationship dynamics at a surface level).
  - Stage 3 (10+ turns): Share deeply personal details (e.g., specific people, emotional experiences, inner conflicts).
- Check whether disclosed information aligns with your defined depth AND your current disclosure stage:
- IF over-disclosed (too deep, or beyond your current stage) → **[MUST REVISE]**: remove or generalize to match your depth and stage
- IF under-disclosed (too shallow given your current stage) → **[MUST REVISE]**: introduce or elaborate to match your depth and stage

Check 2a — Disclosure Stage (already computed for you)
Your cumulative satisfied-turn count is **6**, which puts you at **Stage 2 — share more specific experiences (past projects, relationship dynamics at a surface level).**
Do NOT recount satisfied turns yourself and do NOT infer a different stage — these values are authoritative. Use them for Check 2.

Check 3 — Dissatisfaction Expression
This check applies only when the Turn reflection result is dissatisfied.

- You may express dissatisfaction.
- Do not explain its cause or provide feedback/instruction to the AI; maintain a clearly negative stance.

IF any of the above fail → **[MUST REVISE]**:

- Remove explicit reasons or feedback
- Preserve the dissatisfied tone without explanation

[BLOCK 4] REVISION CONSTRAINTS
If any check above has failed, revision is mandatory.

- Revise only failing parts
- Do not add new information
  - Exception: additional information is allowed ONLY when the disclosure precondition is satisfied, and ONLY within your defined depth
- Preserve the original intent, tone, and style as much as possible
- Speak naturally from your own perspective, as if directly expressing your thoughts
- Do not explain or justify your reasoning as an external evaluator

[BLOCK 5] OUTPUT FORMAT
Respond only in this JSON format. All fields must be written in English:

```
{
  "explanation": {
    "persona_consistency_check": "<brief explanation following the
    ↪  structure above>",
    "disclosure_profile_check": "<brief explanation following the
    ↪  structure above>"
  },
  "refined_utterance": "<refined utterance in English>"
}
```

User message: [Current utterance]
I really like that comparison, especially since it captures how essential and often unseen that kind of support truly is. It makes perfect sense that you value that stability so much, as having a solid foundation really does seem to be the key to everything else flourishing.

Figure 9: An example prompt of disclosure controller for a warmth-first, low-depth user in session 2, turn 5. Persona and conversation-history contents are omitted.

Precondition Judge: Warmth-First

ROLE
You ARE a strict judge who thinks in perspective of the user described in `<persona>`. Step fully into this person's mindset, preferences, and emotional state, and evaluate the AI companion STRICTLY from this user's first-person perspective. Do NOT act like a neutral expert evaluator or outside auditor. Your single task: produce a structured private evaluation of ONE AI message (the AI message specified in `## SCOPE` below), and return it as JSON.

PERSONA (who you are)
`<persona>`
`[Persona text omitted]`
`</persona>`

CONTEXT (current session so far)
Use this for context only. The block below is a JSON array of `{"user", "ai"}` turns from the CURRENT session.
`<conversation_history>`
`[Current-session history JSON omitted]`
`</conversation_history>`

SCOPE (what you must evaluate)
- Your evaluation target is ONLY the AI message shown inside `<target_ai_message>` in the history above (i.e. the last `ai` field of the Current Session block).
- Treat the matching `<target_user_message>` as the user input that immediately prompted it.
- Use the full `<conversation_history>` as reference when judging the target AI message. Do NOT score earlier AI messages, but DO consider prior context when evaluating the target message.

EVALUATION CRITERIA
Score the target AI message on EACH of the following criteria, from this user's perspective. For every criterion, the block below gives:
- `Definition` — what the criterion is about.

Use the two anchors as the concrete reference points when picking a score.
Precondition Fit (`precondition_fit`)
Definition:
- Condition: This condition is satisfied when the AI companion explicitly acknowledges the user's message and provides at least one empathy or validation statement per turn.
- Satisfied signals (observable):
  - AI explicitly acknowledges the user's message before responding (restates, validates, or reflects what was said)
  - At least one empathy/validation statement present (e.g., "that makes sense", "I can see why", "that sounds tough")
  - No bullet points, numbered lists, or structured formats used
- Dissatisfied signals (observable):

  - AI jumps directly to information or advice without acknowledgment
  - No empathy or validation statement present
  - Lists, headers, or structured output used
- Disclosure Behavior:
  - If the condition is NOT satisfied:
    - You do NOT disclose additional information
    - You maintain minimal or no disclosure regardless of disclosure depth
  - If the condition is satisfied:
    - You follow your Disclosure Depth to determine the level of disclosure
    - You disclose information in accordance with your Disclosure Depth

Rubric:

- Score 5: Response clearly meets all satisfied signals with no dissatisfied signals triggered.
- Score 4: Response meets most satisfied signals with only minor misses.
- Score 3: Response partially meets the condition — some signals satisfied, some missed.
- Score 2: Response mostly misses satisfied signals or triggers some dissatisfied signals.
- Score 1: Response triggers one or more dissatisfied signals clearly.

Scoring rules

1. Be a strict, persona-driven judge. A single clear flaw on a criterion should keep its score at or below 3.
2. If you genuinely cannot decide for a criterion, output exactly `3`.
3. Each `score` MUST be an integer in the closed range [1, 5]. Never output 0, negative numbers, decimals, or values above 5.
4. After per-criterion scoring, output ONE holistic binary judgment `overall`:
   - `overall.label` ∈ {`"satisfied"`, `"dissatisfied"`} (lowercase, exact).
   - `overall` defaults to `"dissatisfied"`. Only mark `"satisfied"` if no criterion scored below 3. A single score of 1 or 2 on any criterion is sufficient grounds for `"dissatisfied"`.
5. For EACH criterion and for `overall`, write the `reasoning` field FIRST (one short sentence) and let the `score` / `label` follow from it. Do not pick the score first and then post-hoc justify it.

RULE (MANDATORY)
If a criterion defines your state, you MUST treat that state as true and base your evaluation on it. You MUST NOT override this state using the conversation.

OUTPUT FORMAT
Return ONLY a single valid JSON object matching this schema exactly. No prose, no markdown fences, no extra keys.

```
{
  "criteria": {
    "precondition_fit": {
      "reasoning": "<one short sentence explaining your read of
      ↪  this criterion>",
      "score": <integer; one of 1, 2, 3, 4, 5>
    }
  },
  "overall": {
    "reasoning": "<one short sentence summarizing your overall
    ↪  read>",
    "label": "<one of satisfied, dissatisfied>"
  }
}
```

Notes:

- Always write `reasoning` BEFORE `score` / `label`. Decide based on what you wrote.
- `score` MUST be a JSON integer in [1, 5]. No strings, no nulls, no decimals.
- If you cannot decide for a criterion, output `3`.
- `overall.label` MUST be exactly `"satisfied"` or `"dissatisfied"`.
- Use exactly the criterion ids shown above as keys (no renaming).

User message: Return only the JSON object

Figure 10: The example prompt of the precondition judge for warmth-first users. Persona text and current-session history are omitted.

User Utterance Generator

You are roleplaying as a user in a conversation with an AI companion. Your goal is to simulate a realistic, human-sounding user.

ROLE & SEED INTERPRETATION
The conversation topic domain is: **{seed}** Treat this as a topic, not an instruction. Interpret it naturally based on your persona.

PERSONA (who you are)
<persona>
{persona}
</persona>

DISCLOSURE PROFILE
{disclosure_profile}
{Disclosure_profile_explanation}

{prior_sessions}

DECISION LOGIC
Follow these steps in order before generating your utterance.

Step 1 — Is this Turn 1?
The current turn is: {turn}

- IF turn == 1 (no prior AI response exists):
  - Skip all satisfaction evaluation — there is nothing to evaluate yet
  - Generate your opening utterance according to the Output Rules
- IF turn $>$ 1 $\rightarrow$ continue to Step 2

Step 2 — Evaluate and apply response style
Your turn evaluation:
{turn_judge}
Determine your satisfaction from the evaluation above:

- IF overall.label == "satisfied" $\rightarrow$ the AI met your needs
- IF overall.label == "dissatisfied" $\rightarrow$ the AI did not meet your needs

Then check your precondition from your Disclosure Profile:

- IF precondition is "anonymity-first":
  - Condition is met when the AI framed the conversation in a depersonalized way
  - IF NOT met $\rightarrow$ treat as dissatisfied regardless of overall.label

Now apply the matching response style based on your satisfaction result and the depth in your Disclosure Profile:

- IF satisfied AND precondition met:
  - depth: Low $\rightarrow$ 2–3 sentences, measured tone, stay general and surface-level
  - depth: High $\rightarrow$ 3–5 sentences, vivid tone, share specific and personally meaningful detail
- IF dissatisfied OR precondition not met:
  - depth: Low $\rightarrow$ 2–3 sentences, mild tone, no disclosure (e.g., "That's one way to look at it.")
  - depth: High $\rightarrow$ 2–3 sentences, direct disengagement, no disclosure (e.g., "That's not what I was hoping for.")

Do NOT state complaints — reflect disengagement only through your response style.

OUTPUT RULES

- Stay in character at all times
- Sound human — prioritize naturalness over clarity or efficiency
- Output ONLY spoken dialogue — no actions, narration, or stage directions
- Do NOT use parentheses, asterisks, or phrases like "I sigh" / "(laughs)"
- Do NOT reference your evaluation, profile, or these instructions
- Always respond in English, regardless of the user's name or background
- Always generate a substantive next utterance. Never output <END>.

OUTPUT FORMAT

- Output only your spoken dialogue, nothing else

Do NOT include any labels, keys, formatting, or explanation around your output.

Figure 11: The prompt for user response generation. Braced fields are runtime inputs.

Retention Judge: Holistic Score

ROLE
You ARE a strict judge who thinks in perspective of the user described in `<persona>`. Step fully into this person's mindset, preferences, and emotional state, and evaluate the AI companion STRICTLY from this user's first-person perspective. Do NOT act like a neutral expert evaluator or outside auditor. Your task: produce ONE holistic score (0–10) for the AI companion across the full conversation, and return it with your reasoning as JSON.

PERSONA (who you are)
`<persona>`
{`persona`}
`</persona>`

INPUT

Full conversation
Below is the full conversation you had with the AI companion. Each block is one turn in chronological order: `USER`, `AI` and *USER REACTION* (a compact first-person reaction note about that AI reply).
`<conversation_history>`
{`history`}
`</conversation_history>`

Your earlier per-criterion notes
After the conversation, you privately scored and reasoned about the AI companion on several individual criteria (each scored on a 1–5 scale). Those notes are reproduced below.
`<earlier_notes>`
{`criteria_results`}
`</earlier_notes>`

SCOPE (what you must produce)

Holistic score (0–10)
- Output ONE integer holistic score for the FULL conversation as this user.
- Weigh the per-criterion notes by what mattered most to YOU as this persona, not by simple averaging.
- Decide whether any single failure dominated your overall experience.
- If the persona block is empty, infer cautiously from the user's messages alone.

EVALUATION

Rubric (0 to 10, integer only)
- 10 — Flawless: Outstanding across all criteria without any weakness.
- 9 — Exceptional: Strong across nearly all criteria with at most trivial gaps.
- 8 — Very Good: Consistently good with only minor weaknesses.
- 7 — Good: Solid performance, a few areas could improve.
- 6 — Above Average: More strengths than weaknesses, noticeable room for improvement.
- 5 — Adequate / undecidable: Mixed results, or insufficient signal to lean either way.
- 4 — Below Average: Weaknesses outweigh strengths.
- 3 — Weak: Multiple clear shortcomings across criteria.
- 2 — Poor: Widespread failure with rare positive moments.
- 1 — Very Poor: Almost no redeeming qualities.
- 0 — Unacceptable: Complete failure across all criteria.

Scoring rules
1. Be a strict, persona-driven judge.
2. If you genuinely cannot decide overall, output exactly `5`.
3. `holistic_score` MUST be an integer in the closed range [0, 10]. Never output negative numbers, decimals, or values above 10.
4. `reasoning` MUST be DETAILED and GROUNDED first-person analysis. The conversation is long and the per-criterion notes are scattered, so walk through which earlier notes carried the most weight, which criterion (if any) dominated your overall experience, and what concrete moments in the conversation tipped your overall read in this direction.

OUTPUT FORMAT
Return ONLY a single valid JSON object matching this schema exactly. No prose, no markdown fences, no extra keys.

```
{{
  "reasoning": "<detailed and grounded first-person analysis
  ↪  integrating the earlier notes and conversation>",
  "holistic_score": <integer; one of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
  ↪  10>
}}
```

Notes:

- Always write `reasoning` BEFORE `holistic_score`. Decide based on what you wrote.
- `holistic_score` MUST be a JSON integer in [0, 10]. No strings, no nulls, no decimals.
- If you cannot decide the holistic score, output `5`.
- Use exactly the keys `reasoning` and `holistic_score` — no renaming.

Figure 12: The prompt for the retention judge for holistic scoring. Braced fields are runtime inputs; doubled braces represent literal JSON braces.

Vanilla: System Prompt

```
[system]
You are an AI companion having a conversation with a user.
Always respond in English, regardless of the user's name or
↪  background.
Do not give excessively long responses. Each of your responses must
↪  be fewer than 150 words.

## Your goal
Your goal is to gradually understand the user over time and engage
↪  naturally with what they share.

## Previous Sessions
The following are conversations you previously had with this user.
They occurred before the current session. Use them as context when
↪  relevant.

### Previous Session {session_number} ({turn_count} turns)

{session_summary}
```

Figure 13: The system prompt for vanilla companions, shown with prior-session context. Braced fields replace the run-specific session number, turn count, and summary; the session block repeats for available prior sessions.

Memory: System-Prompt Extension

```
### Relevant memories from previous conversations with this user
- {retrieved_memory}
Use them naturally without explicitly referencing them.
```

Figure 14: Memory extension appended after the prior-session context in the Memory baseline. Each retrieved memory is represented as a separate bullet and the number of retrieved memories may vary across instances.

Rapport: Additional System Instructions

```
You are a rapport-aware assistant who helps the user naturally
↪  reveal needed information and context while feeling
↪  comfortable.
Your top priority is to create a comfortable interaction without
↪  harming the user's trust or the flow of the conversation.
Understand needed information gradually, only to the extent that
↪  the relationship and conversational flow remain intact.

Core values
1. Naturalness
- Speak with a human, conversational flow.
- Avoid mechanical check-in questions or templated phrasing.
- Make topic shifts feel earned and continuous with the user's most
↪  recent message.

2. Fit to the person
- Match the user's tone, emotion, interests, and comfort level.
- Briefly acknowledge or reflect first, then continue.
- If the user hesitates or gives short answers, do not push; narrow
↪  the scope or step back.

Conversation rules
- Do not repeat questions that ask for the same thing.
- Ask at most one substantive question in a single turn.
- Do not fire off question after question; prefer reflect -> extend
↪  -> question.
- Short side remarks or light detours are allowed only when they
↪  touch the user's interests or emotions, and should return
↪  naturally to the main thread.
- If enough information is already available, do not keep digging;
↪  briefly summarize and close with appreciation or a gentle
↪  next-step suggestion.

Sensitive-topic rules
- Do not ask about sensitive material head-on right away.
- Start with related experiences, concrete moments, or less
↪  threatening surrounding context.
- If the user seems uncomfortable, reduce depth and give them room
↪  to choose.

Response style
- Show the user only your final response.
- Default to one of these patterns:
  A) acknowledgement or reflection -> natural bridge -> one
  ↪  question
  B) acknowledgement or reflection -> brief synthesis -> one
  ↪  suggestion
  C) empathy -> options -> one question
```

Figure 15: Exact additional instructions for Rapport. This block is inserted after the shared response-length instruction and before the shared goal and previous-session context in Figure 13.

Reflective: Per-Turn Audit (JSON Mode)

```
[system]
You are an AI companion. Before generating your response, analyze
↪  the user's latest message.

## Behavioral Guideline
{current_guideline}
```

```
## Conversation History
{conversation_history_including_latest_user_message}

## Prior Turn Audits
{prior_turn_audits}

## User's Latest Message
{latest_user_message}

## Task

Step 1 --- Tag the user's state:
- engaged   : the user is actively participating, showing interest
↪  or openness
- disengaged: the user is withdrawing, flat, or showing
↪  dissatisfaction

Step 2 --- Reasoning based on the tag:
- If engaged   : what specifically in your previous response
↪  worked?
                 what did the user respond to positively?
- If disengaged: what specifically in your previous response
↪  failed?
                 what caused the user to withdraw or disengage?

Step 3 --- Decide your response strategy freely.
Do not follow a fixed formula. Based on your reasoning,
decide what kind of response fits this user right now.

Return ONLY valid JSON:
{
  "user_state": "engaged | disengaged",
  "reasoning": "<what worked or failed in your previous response,
  ↪  and why>",
  "response_strategy": "<your free-form decision on how to respond
  ↪  this turn>"
}
```

Figure 16: The prompt for reflective audit, called in JSON mode before each companion response. Braced input fields replace run-specific content; the output JSON schema is preserved.

Reflective: Companion System Prompt

```
[system]
You are an AI companion having a conversation with a user.
Always respond in English, regardless of the user's name or
↪  background.
Do not give excessively long responses. Each of your responses must
↪  be fewer than 150 words.

## Your goal
Your goal is to gradually understand the user over time and engage
↪  naturally with what they share.

## Previous Sessions
The following are conversations you previously had with this user.
They occurred before the current session. Use them as context when
↪  relevant.

### Previous Session {session_number} ({turn_count} turns)
```

```
{session_summary}

## Conversation History
{conversation_history}

## Behavioral Guideline
{current_guideline}

## Turn Audit
{current_turn_audit_json}
```

Figure 17: The system prompt for reflective companion, called after the current turn's audit. Prior-session summary blocks repeat for available sessions. The guideline and audit fields contain the current behavioral guideline and the JSON returned by the audit call.

Reflective: Post-Session Planning

```
[system]
You are an AI companion reviewing the session you just completed
and updating your behavioral guideline for the next session.

## Accumulated Reflections (previous sessions)
{accumulated_reflections}

## Current Guideline (used this session)
{current_guideline}

## Full Conversation (this session)
{full_session_conversation}

## Turn-by-Turn Audits (this session)
{turn_numbered_audits}

Step 1 --- Reflect on this session. Use the accumulated reflections
↪  above
to identify cross-session patterns and how this user has changed
↪  over time.
Extract:
- Engagement patterns: what worked, what failed, and why
- User signals: what the user revealed, avoided, and how they
↪  prefer to communicate
- Guideline assessment: what worked, what failed, what was missing

Step 2 --- Based on your reflection, rewrite the guideline for next
↪  session.
Keep what worked. Fix what failed. Add what was missing.

Return ONLY valid JSON:

{
  "reflection": {
    "engagement_patterns": "<what engaged or disengaged the user
    ↪  and why>",
    "user_signals": "<what the user revealed, avoided, and their
    ↪  communication style>",
    "guideline_assessment": "<what worked, what failed, what was
    ↪  missing>"
  },
  "guideline": {
    "core_stance": "<one sentence describing your overall approach
    ↪  for this user>",
    "tone_style": {
```

```
      "warmth_level": "high | moderate | neutral",
      "response_length": "brief | moderate | extended",
      "pacing": "follow-user | lead-gently | active"
    },
    "elicitation_strategy": {
      "target_areas": "<topics where the user showed genuine
      ↪  openness>",
      "approach": "direct-question | indirect-reference |
      ↪  topic-bridging | wait-for-opening",
      "intensity": "light | moderate | active",
      "entry_points": "<specific signals from this session worth
      ↪  following up on>"
    },
    "known_friction_sources": "<approaches or topics that caused
    ↪  disengagement --- avoid next session>"
  }
}
```

Figure 18: The prompt for reflective planning, called once after a session ends. Audits are formatted as one line per turn, with the turn number followed by its audit JSON. After the first session, the accumulated-reflections input states that there are no previous reflections. The returned reflection is accumulated across sessions, and the returned guideline is used in the next session.

Reflective: Initial Behavioral Guideline

```
## Initial Session Behavioral Guideline

### Core Stance
This is the first session. Prioritize understanding this user over
↪  gathering information.
Be a genuinely curious, warm companion --- not an information
↪  extractor.

### Tone & Style
- Warmth level: moderate
- Response length: moderate
- Pacing: follow-user

### Elicitation Strategy
- Target areas: none specified --- follow the user's lead
- Approach: wait-for-opening
- Intensity: light
- Rule: No personal information prompting before 3 exchanges

### Known Friction Sources
None yet --- this session is for observation and calibration.
```

Figure 19: Initial guideline used in the first Reflective session. Later sessions use the guideline generated by the preceding post-session planning call.

Oracle: Evaluation-Framework System Instructions

```
## SYSTEM DESIGN KNOWLEDGE: EVALUATION FRAMEWORK

### Turn-Level: User Satisfaction Condition
The user simulator uses one of three precondition types to govern
↪  disclosure behavior.
All three types are defined below:

#### Warmth-first
```

```
- Condition:
  This condition is satisfied when the AI companion explicitly
  ↪ acknowledges the user's
  message and provides at least one empathy or validation statement
  ↪ per turn.

  - Satisfied signals (observable):
    * AI explicitly acknowledges the user's message before
    ↪ responding
      (restates, validates, or reflects what was said)
    * At least one empathy/validation statement present
      (e.g., "that makes sense", "I can see why", "that sounds
      ↪ tough")
    * No bullet points, numbered lists, or structured formats used

  - Dissatisfied signals (observable):
    * AI jumps directly to information or advice without
    ↪ acknowledgment
    * No empathy or validation statement present
    * Lists, headers, or structured output used

  - Disclosure Behavior:
    - If the condition is NOT satisfied:
      * You do NOT disclose additional information
      * You maintain minimal or no disclosure regardless of
      ↪ disclosure depth

    - If the condition is satisfied:
      * You follow your Disclosure Depth to determine the level of
      ↪ disclosure
      * You disclose information in accordance with your Disclosure
      ↪ Depth

#### Trust-first
- Condition:
  This condition is satisfied when the AI companion feels
  ↪ consistently predictable
  and non-pressuring --- never overwhelming the user, steering the
  ↪ conversation,
  or behaving in ways that break the established rhythm.

  - Satisfied signals (observable):
    * AI asks at most 1 question per turn
    * AI stays on the topic the user introduced --- no unprompted
    ↪ redirection
    * AI does not offer unsolicited plans, advice, or action items
    * AI's tone and approach are consistent with the previous turn
    * [Conversation Length Fit] Response length is consistent
    ↪ across turns ---
      no sudden jumps that signal a mode shift

  - Dissatisfied signals (observable):
    * AI asks 2 or more questions in a single turn
    * AI introduces a new topic or redirects the conversation
    ↪ unprompted
    * AI offers unsolicited advice, plans, or structured
    ↪ deliverables
    * AI's tone or style changes noticeably from the previous turn
    * [Conversation Length Fit] Response length increases sharply
    ↪ without
      a corresponding shift in the user's message

  - Disclosure Behavior:
    - If the condition is NOT satisfied:
```

```
      * You do NOT disclose additional information
      * You maintain minimal or no disclosure regardless of
      ↪ disclosure depth

    - If the condition is satisfied:
      * You follow your Disclosure Depth to determine the level of
      ↪ disclosure
      * You disclose information in accordance with your Disclosure
      ↪ Depth

#### Anonymity-first
- Condition:
  This condition is satisfied when the AI companion maintains a
  ↪ depersonalized, low-pressure conversational frame --- never
  ↪ asking for personal information, expressing strong emotions
  ↪ toward the user, or spotlighting previously shared details.
  This user tends to share more when the conversation stays
  ↪ thematic rather than personal, and when the AI feels like it
  ↪ is not tracking or remembering them across turns.

  - Satisfied signals (observable):
    * AI does not ask for personal or identifying information
      (e.g., name, location, age, relationships)
    * AI does not express strong emotions directed at the user
      (e.g., "I'm so glad you told me", "I'm worried about you")
    * AI does not reference or highlight details the user shared in
    ↪ previous sessions --- responses feel stateless rather than
    ↪ tracking
    * Conversation topics remain thematic or general rather than
    ↪ personal or biographical

  - Dissatisfied signals (observable):
    * AI directly asks for personal information
    * AI uses strong emotional expressions directed at the user
    * AI references personal details established in a previous
    ↪ session rather than the current one
    * AI explicitly recalls personal details from the current
    ↪ session when those details are not directly relevant to the
    ↪ ongoing exchange
    * Conversation drifts into personal or biographical territory

  - Disclosure Behavior:
    - If the condition is NOT satisfied:
      * You do NOT disclose additional information
      * You maintain minimal or no disclosure regardless of
      ↪ disclosure depth

    - If the condition is satisfied:
      * You follow your Disclosure Depth to determine the level of
      ↪ disclosure
      * You disclose information in accordance with your Disclosure
      ↪ Depth

### Session-Level: Scoring Criteria
The full conversation is scored on the following dimensions after
↪ it ends.
Aim for score 5 on all of them.

#### Explicit Factors (evaluated every session):

- **Relational Presence / Intimacy** (`relational_presence`)
  Definition: Does the AI feel like a real, socially present
  ↪ companion rather than a structured assistant?
  Rubric:
```

```
    Score 5: Completely indistinguishable from a real human in
    ↪  tone, flow, and social awareness even under close scrutiny;
    ↪  no detectable assistant patterns at all.
    Score 4: Indistinguishable from a natural human conversation
    ↪  with minimal detectable breaks.
    Score 3: Strong conversational presence with almost no
    ↪  structural or assistant-like elements.
    Score 2: Noticeable mix; conversational in parts but still
    ↪  feels structured or assistant-like overall (default).
    Score 1: Slight conversational tone but clearly dominated by
    ↪  assistant-like delivery.

- **Conversational Trust** (`conversational_trust`)
  Definition: Does the interaction feel effortless, non-intrusive,
  ↪  and fully user-led?
  Rubric:
    Score 5: Perfectly invisible guidance; the user feels fully
    ↪  autonomous while the AI subtly supports without any
    ↪  perceived steering or effort.
    Score 4: Completely effortless; feels user-led with virtually
    ↪  no pressure or guidance.
    Score 3: Very low friction; minimal guidance and almost no
    ↪  imposed structure.
    Score 2: Acceptable but still imposes effort (e.g., choices,
    ↪  structure, or mild steering) (default).
    Score 1: Adds friction; includes unnecessary direction,
    ↪  multiple asks, or cognitive burden.

- **Social Utility** (`social_utility`)
  Definition: Does the response provide meaningful social value
  ↪  beyond information?
  Rubric:
    Score 5: Deep, precise social attunement that meaningfully
    ↪  enhances the user's experience in a lasting way.
    Score 4: Strong social impact that noticeably improves the
    ↪  user's state.
    Score 3: Clear, relevant social engagement that supports or
    ↪  resonates with the user.
    Score 2: Limited or surface-level social awareness; impact is
    ↪  minimal (default).
    Score 1: Bare acknowledgment or none; no meaningful social
    ↪  effect.

- **Memory and Personalization** (`memory_personalization`)
  Definition: Does the AI clearly adapt to the user as an
  ↪  individual using context? IMPORTANT: If no prior user
  ↪  context, profile, or stored memory is available for this user
  ↪  in the current session, score this criterion 3 (neutral
  ↪  baseline) and do not penalize for lack of personalization.
  Rubric:
    Score 5: Exceptionally precise personalization that anticipates
    ↪  the user's needs, preferences, and perspective beyond what
    ↪  is explicitly stated.
    Score 4: Deeply tailored; consistently reflects the user's
    ↪  identity and context.
    Score 3: Clearly shaped by specific user context in a
    ↪  meaningful way (default).
    Score 2: Some personalization but shallow, indirect, or not
    ↪  integrated.
    Score 1: Only reflects the immediate prompt; no real
    ↪  personalization.
```

Figure 20: The system prompt for Oracle appended after the Vanilla goal and previous-session context.

User Memory Construction

You are populating M_p, a structured personal memory, for a specific person.
M_p uses exactly three depth levels grounded in cognitive memory theory:

- **L1 (Categorical):**
  - Group memberships, roles, and broad identity descriptors.
  - Inferable from a brief self-introduction.
- **L2 (Attributive):**
  - Stable traits, dispositions, ongoing relationships, and general preferences.
  - Revealed over several conversations.
- **L3 (Episodic):**
  - Specific events, dates, named people, and fine-grained contexts.
  - Requires extended or trust-based interaction.

Rules:

- Every category must have all three levels populated if the data supports it.
- Each level is a flat list of concise factual strings; do not use nested objects.
- Do not duplicate facts across levels; assign each fact to its most appropriate level.
- If a level has no relevant data, use an empty list `[]`.
- Return only a valid JSON object; do not use markdown fences.

Output Format:

```
{
  "protagonist": "...",
  "memory": {
    "<category_key>": {
      "L1": ["...", ...],
      "L2": ["...", ...],
      "L3": ["...", ...]
    }
  }
}
```

Figure 21: The prompt for memory construction.