Mind2Dialogue: Training Human-Aware Language Models by Simulating User Mental States

Zixuan Wang[1,*] **Yufan Zhou**[2*] **Jinzhou Tang**[1*] **Chengjun Wu**[1] **Xinle Yu**[1]
Lyumanshan Ye[1] **Zhaoxiang Feng**[1] **Letian Peng**[1] **Adyasha Patra**[1]
Fan Bai[5] **Enze Ma**[3] **Zhengding Hu**[1] **Jianyang Gu**[4] **Zhao Wang**[1]
Yufei Ding[1] **Jingbo Shang**[1] **Tianmin Shu**[5] **Zhiting Hu**[1] **Zhen Wang**[1,‡]
[1]UC San Diego [2]KU Leuven [3]University of Illinois Chicago
[4]The Ohio State University [5]Johns Hopkins University
ziw178@ucsd.edu, zhw085@ucsd.edu

Project Code Dataset

Abstract

Language models that understand people's evolving mental states could become long-term partners in learning, reasoning, and decision-making. Training such *human-aware* models faces a supervision gap because conversational data often omit the mental states that explain user behavior. We propose Mind2Dialogue, an integrated framework that connects mental-state simulation with assistant learning. M2D-SIM gives a user simulator and an Oracle assistant access to the same evolving state, linking the beliefs and goals that drive user behavior to the responses used for supervision. The resulting M2D-CORPUS combines dialogues and question-answer examples to train M2D-CHAT through distillation, with the evolving state withheld from the student. Across Qwen, Llama, and OLMo, training improves every reported personalization metric, including gains of 26.6 to 40.9 percentage points in preference-following generation. The same training improves belief and action reasoning on Qwen and Llama; Llama-3.1-8B gains 24.8 points on BigToM forward-belief accuracy and 17.8 points on forward-action accuracy. Mind2Dialogue makes simulated user states a practical source of supervision for training language models to understand and assist people.

1 Introduction

Language models should help people learn, reason, and make decisions by accounting for the beliefs, goals, and circumstances that shape their actions. Understanding others is a core component of human intelligence and motivates *human-aware* capabilities in the next generation of language models [5]. In real-world deployment, useful assistance must reflect the knowledge, priorities, and constraints of the person using the model [4, 5]. Sustained human-aware assistance requires adapting to *evolving user states* as beliefs, goals, and circumstances change.

Training human-aware language models faces a supervision gap. Conversations between people who know one another well provide natural examples of assistance informed by mutual understanding. Close friends, family members, and longtime collaborators draw on *shared experience* to recognize the goals behind a request and respond with knowledge of the person's circumstances [4]. Documenting such exchanges and their shared background raises privacy and consent concerns and requires substantial annotation effort [30, 17]. Publicly available conversations can preserve the words exchanged without revealing the beliefs or goals behind a response [12]. The missing supervision connects a person's *evolving state* to the behavior it produces and an informed partner's response.

*Equal contribution. ‡Corresponding author.

Preprint.

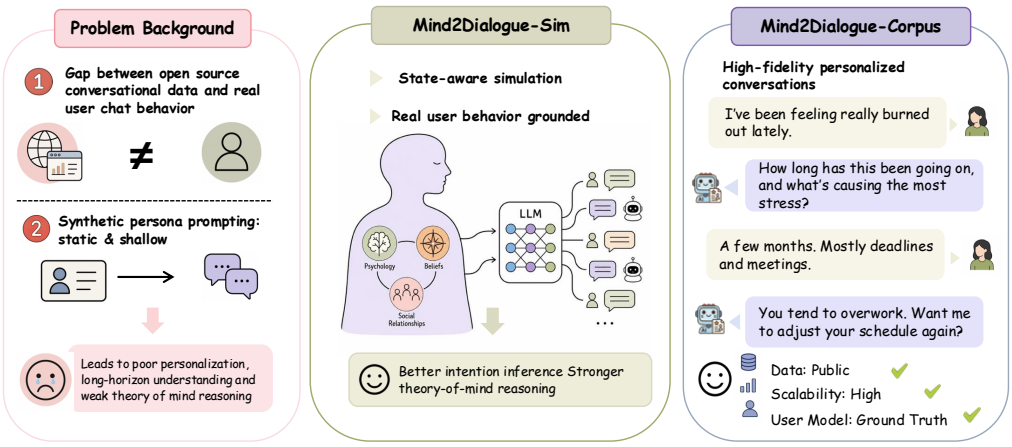

Figure 1: Training human-aware language models. Mind2Dialogue uses simulated mental states to teach assistants to respond to the beliefs, goals, and needs behind a user's words. A shared state drives user behavior and informs Oracle responses; students learn these demonstrations with the state withheld. "Ground Truth" denotes the simulator-defined user state.

Synthetic data has emerged as a promising solution to this supervision gap [11, 30]. Persona-conditioned generation uses descriptions of users' backgrounds and preferences to diversify synthetic conversations [11, 17]. Profile-based generation supplies a consistent identity, but a static description leaves changes in the user's beliefs, goals, and emotions implicit in the generated exchange. A more recent line of work casts an LLM as a goal-driven user and pairs it with an off-the-shelf assistant to generate multi-turn and multi-session dialogues at scale [33, 8]. State modeling and simulated feedback further improve user fidelity and assistant adaptation [53, 22, 58, 29]. Yet realistic user simulation does not ensure that assistants understand their users. Evaluations with UserLM and LifeSim document failures to interpret and act on users' implicit intentions [33, 8]. When an assistant generates demonstrations from incomplete dialogue, the target responses depend on its own uncertain interpretation of the user. Distillation passes that interpretation to the student, leaving its supervision dependent on the teacher's existing ability to understand the user.

In this paper, we propose Mind2Dialogue to close this supervision gap by simulating the mental states behind user behavior. To construct demonstrations informed by the user's state, we introduce an Oracle assistant with direct access to that state during generation. The key idea is *shared-state user simulation*, which uses one evolving state to generate user behavior and guide the Oracle's responses. The Oracle can thus demonstrate how to assist a user whose beliefs, goals, and emotions are only partially expressed in the dialogue. Specifying the state before generating each turn makes the connection between user behavior and informed assistance available by construction.

Scaling mental-state supervision requires diversity across users and coherence within each interaction. We therefore build M2D-SIM with scenarios that give users reasons to seek assistance, state updates that track their changing circumstances, and a controller that varies their conversational behavior (Figure 2). The resulting M2D-CORPUS combines multi-turn Oracle dialogues with question-answer examples derived from the same interactions, without requiring human annotation for each generated dialogue. To transfer the Oracle's decisions to a deployable model, we distill these demonstrations into M2D-CHAT through supervised fine-tuning [26]. The student learns from the visible inputs and target responses, with the evolving state withheld during both training and inference. This information asymmetry allows mental states to guide what the model learns without requiring those states as inputs when the model assists a user.

However, evaluating human-aware language models through long-term interaction with real users is difficult to scale [25]. We propose an evaluation that connects two seemingly distinct domains, personalization and theory of mind. We test whether the same training improves personalized responses and reasoning about other agents' beliefs and actions. Training on M2D-CORPUS improves every measured personalization metric on PersonaMem-v1, PersonaMem-v2, and PrefEval [18, 19, 57] across Qwen2.5-7B, Llama-3.1-8B, and OLMo-3-7B (Table 5). Qwen2.5-7B gains 33.4 percentage

points on PrefEval generation and 10.0 points on PersonaMem-v2 multiple-choice accuracy over its base model (Table 1). On ToMi and BigToM [23, 10], Qwen and Llama improve across all three tasks, including a 13.0-point gain for Qwen on BigToM forward-belief accuracy (Tables 2 and 5). OLMo improves on ToMi and declines on both BigToM tasks, showing that the benefits for mental-state reasoning vary across models. Mind2Dialogue establishes an integrated framework for training human-aware language models, demonstrating how simulating user mental states can advance both personalized assistance and reasoning about other agents.

2 Related Work

Personalization and user modeling. Personalized language models augment a fixed model with external memory and retrieval [3, 59, 24] or adapt model parameters using user-specific data [43, 31]. PersonaMem-v2 uses preference supervision for reinforcement fine-tuning and agentic memory learning [19]. DreamCUB learns a dialogue world model that predicts utterances and user beliefs for model-based reinforcement learning [58]; PUMA maintains beliefs over partially observed user states and plans using predicted state transitions [29]. Mind2Dialogue constructs assistant supervision by giving the teacher direct access to the simulated state that generates user behavior.

Synthetic dialogue and user simulation. Synthetic dialogue research ranges from fixed persona-conditioned generation [55, 11] to multi-turn and multi-session LLM user simulators [6, 44, 33, 19]. Persona Hub expands profile diversity, while Synthetic-Persona-Chat improves persona consistency through generation and critique [11, 17]. Generative Agents and LifeSim extend simulation to behavior shaped by memory and changing circumstances [37, 8]. UserLM learns intent-conditioned user behavior from human conversations [33], and HumanLM aligns generated mental states and responses with real users through reinforcement learning [53]. HumanLM trains the simulator; Mind2Dialogue uses simulation to train the assistant.

Simulated interaction also supports assistant learning through feedback and rewards. ProPerSim adapts proactive recommendations using simulated user ratings [22]; PersonaGym supports personalized prompt optimization through profile inference and outcome feedback [30]. UserRL trains interactive agents with simulated users and studies turn-level rewards and trajectory scoring [39]. We focus on *who observes the simulator-defined user state when assistant responses are generated*. The Oracle observes the state that drives user behavior before generating the response target.

Social intelligence and mental-state reasoning. Social intelligence research examines both inferring other agents' states and using that understanding to act. Machine Theory of Mind learns to predict agents' behavior and mental states from observed trajectories [41]; SOTOPIA-π trains language agents through behavior cloning and self-reinforcement on interactions selected by social-goal ratings [51]. SimpleToM shows that accurate state attribution can coexist with errors in predicting or judging behavior [14], motivating our joint assessment of assistance and mental-state reasoning.

Mind2Dialogue uses mental states to supervise assistant decisions, while ToMATO uses them as targets for mental-state inference [47]. ToMATO combines personas with turn-level first- and second-order thoughts, keeping each speaker's thoughts hidden from its partner. Those thoughts supply mental-state QA labels for evaluation and for fine-tuning on separately generated conversations. Our Oracle observes the evolving state that produces the user's behavior and uses it to generate response demonstrations for a student with that state withheld. This state access lets the teacher demonstrate decisions whose rationale is only partly visible to the student.

Learning with privileged information. Learning with privileged information and generalized distillation allow a teacher to use information unavailable to the student [50, 15, 26]. Whereas contemporaneous work uses joint or on-policy objectives to transfer from privileged policies [38], we rely on a fixed Oracle and standard supervised fine-tuning. The privileged information in Mind2Dialogue is the state that generates the interaction itself. Sharing this state with the Oracle ties the teacher's additional knowledge to the causes of user behavior, while the student receives only observable inputs and target responses. Our contribution is the construction of this supervision through an integrated simulator, corpus, and assistant-training pipeline.

3 The Mind2Dialogue Framework

Mind2Dialogue is a framework for training human-aware language models with supervision informed by simulated user states. Mind2Dialogue generates multi-turn training data with a user simulator and an Oracle assistant. During data generation, both components have access to a persona p and a shared structured state s_t. M2D-SIM is the simulator that generates these interactions. M2D-CORPUS contains these dialogues and derived question-answer examples. M2D-CHAT denotes the student language models trained on this corpus with the evolving state withheld. This design separates information used to generate the data from information available to the student.

3.1 Problem Formulation and Shared-State Supervision

Problem formulation. Human-aware training must teach assistants to respond to users whose beliefs, goals, and circumstances are only partly expressed in conversation. Let p denote the persona used to generate a dialogue, $H_{<t}$ the history of the dialogue before the t-th user message and m_t the user message at that turn. We define

$$H_t = (H_{<t}, m_t)$$

as the dialogue history through the user message. Let s_t represent the user's evolving state and x_t the student input, which includes H_t and excludes s_t. The learning objective is an assistant policy $\pi_\phi(a_t \mid x_t)$ that uses the available evidence about the user's state to guide its responses.

The supervision gap arises when the response target depends on a user state that the input does not fully specify. A teacher restricted to x_t must infer the missing state before choosing a response, making supervision depend on the teacher's existing understanding of users. We seek demonstrations whose targets are generated with knowledge of s_t, while retaining x_t as the student's input.

Shared-state Oracle supervision. Mind2Dialogue addresses this problem by generating the user state and the dialogue together. The key idea is to share that state with the assistant before it produces a training target. We use *Oracle* to denote direct access to the simulated state during response generation. A common persona alone does not specify how beliefs and goals change within the interaction; sharing s_t gives both policies the same evolving account of those changes. At each turn, a state-update policy first produces the next state, a user policy produces the next user message, and an Oracle policy produces the target response from the updated dialogue history.

$$\begin{aligned} s_t &\sim \pi_{\text{user}}(\cdot \mid p, s_{t-1}, H_{<t}), \\ m_t &\sim \pi_{\text{user}}(\cdot \mid p, s_t, H_{<t}), \\ a_t &\sim \pi_{\text{oracle}}(\cdot \mid p, s_t, H_t). \end{aligned}$$

The shared s_t connects the cause of the simulated behavior with the information used to produce its response. The next state update observes $H_{<t+1} = (H_t, a_t)$, including the assistant response. Subsequent turns can therefore reflect changes induced by the assistant's response.

From N generated dialogues, we construct the dialogue training set

$$\text{M2D-CORPUS}_{\text{dialogue}} = \left\{ \left(x_t^{(i)}, a_t^{(i)}\right) \;\middle|\; 1 \le i \le N,\ 1 \le t \le T_i \right\},$$

where T_i is the number of assistant-response targets in dialogue i. Each response target is thus paired with the evidence the student will observe, while its construction also uses the underlying simulated state. Sections 3.2 and 3.3 describe how we control these trajectories and retain coherent demonstrations; Section 3.4 formalizes learning with the state withheld.

3.2 Controlled Stateful User Simulation

User behavior reflects the interaction between personal characteristics and situational demands [9]. The cognitive-affective processing account further describes how situational features interact with goals, affect, expectations, and related internal variables to produce context-dependent behavior [32]. In our implementation, the scenario supplies the immediate context for the interaction, s_t records simulator-defined variables that carry across turns, and dynamic behavior-mode prompting controls how the user acts at each turn. If these elements are left implicit in a single prompt, the LLM must determine all three during generation. M2D-SIM makes these controls explicit to preserve continuity while varying the situations and behaviors represented in the training data.

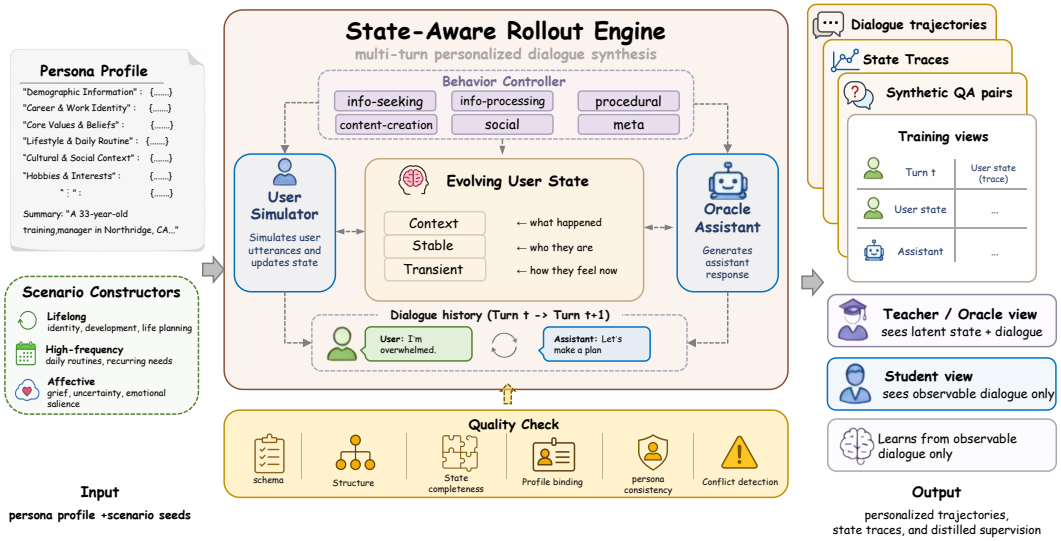

Figure 2: Shared-state user simulation. M2D-SIM generates user behavior and Oracle responses from one evolving state. A persona and scenario establish the context; state updates preserve continuity, and behavior guidance varies how the user communicates. Quality checks select trajectories for dialogue and QA supervision. State traces guide generation and are withheld from student inputs.

Persona-grounded scenario construction. The scenario specifies why the user starts the interaction and provides a setting in which persona-specific information can affect how the assistant should respond. Each dialogue begins with a scenario derived from the persona p. We construct scenarios in three categories: LIFELONG, covering identity and long-term personal development; HIGH-FREQUENCY, covering recurring everyday needs; and AFFECTIVE, covering emotionally significant situations such as grief or uncertainty. We filter candidate scenarios using three criteria: level of abstraction, embedding-based distinctness from existing scenarios, and semantic consistency with the persona. For the distinctness check, we reject a candidate if its cosine similarity to any existing scenario exceeds θ_{sim}. Accepted scenarios are cached for each persona so that subsequent runs can reuse the same scenario set.

Evolving user-state simulation. An assistant's response can change what a user believes or needs, so the simulator must carry those changes into subsequent turns. M2D-SIM represents the state shared by the generation policies as a structured record:

$$s_t = \left(c_t,\ z_t^{\text{stable}},\ z_t^{\text{transient}}\right),$$

where c_t tracks the turn index, unresolved goals, trust history, and a summary of the interaction. The component z_t^{stable} stores information that changes slowly, such as user values, background constraints, and the user's position toward the assistant. The component $z_t^{\text{transient}}$ stores short-lived information, such as mood, current concerns, and judgments made during the turn. These fields are simulator-defined control variables, not measurements of a real user's mental state.

Turn-level behavior control. Users can express a goal through questions, requests, or reactions to an assistant, so varied interactions also require control over conversational behavior. To vary user behavior in a dialogue, we use a behavior controller based on the Taxonomy of User Needs and Actions (TUNA) proposed by Shelby et al. [46]. At each turn, the controller selects one of 16 behavioral modes from six families: information seeking, information processing, procedural guidance, content creation, social interaction, and meta-conversation. It then adds the instructions associated with the selected mode to the user-simulator prompt. It provides less behavioral guidance at turn 1 and more guidance at later turns or on turns with greater delegation to the assistant. Its selection procedure also encourages coverage across the six families. The full taxonomy and mode descriptions appear in Section H.

3.3 State-Informed Training Corpus Construction

A simulated interaction supplies both examples of informed assistance and the context needed to ask questions about the user. The dialogue view pairs the student-visible context with the Oracle response. The QA view uses the persona, saved state trajectory, and a dialogue excerpt to generate questions and answers about the simulated interaction. We generate multiple-choice persona-memory examples and free-form preference-following examples using the same message schema as the dialogue examples. The mixture also contains preference-classification questions. Student QA inputs contain the observable context, the question, and any answer options required by the format; the structured state remains available only during generation. Section F.5 reports the composition of the training mixture.

Training requires coherent trajectories that preserve the simulated user's context. We apply four programmatic checks to generated conversations: schema validity, structural sanity (turn count, role alternation, and token bounds), state-trajectory completeness, and profile binding. Two LLM-based checks additionally score persona consistency and flag conflicts with fixed persona attributes, using a judge model distinct from the simulator and the Oracle. Section F.6 describes the checks and reports a human audit of the filtered corpus; the two judge rubrics are represented in Section G.5.

3.4 Privileged-State Supervised Distillation

Human-aware assistance requires learning response decisions under partial observability of the user's state. We use the Oracle's state access to construct targets while keeping the student's information constraints identical at training and inference. The setup follows generalized distillation [26], with privileged information supplied through response targets. Direct state access supplies additional supervision even when the Oracle and user simulator share a backbone.

The student learns the Oracle's response behavior through the evidence available in its input. For dialogue demonstrations, let $q(x, s, y)$ denote the joint distribution of student inputs, simulated states, and Oracle responses induced by simulation and quality filtering. Conditioning on the student's input gives the response distribution

$$\bar{q}(y \mid x) = \mathbb{E}_{s \sim q(\cdot \mid x)}[q(y \mid x, s)]\,. \tag{1}$$

The conditional $q(s \mid x)$ accounts for states consistent with the visible context, and $q(y \mid x, s)$ captures the corresponding state-informed targets. These distributions describe the generated data and are not separately estimated during training.

We fine-tune a student policy π_ϕ on M2D-CORPUS. Each training example consists of a student-visible input sequence x and a target assistant response y.

We minimize the standard autoregressive cross-entropy loss over the target response tokens:

$$\mathcal{L}_{\mathrm{SFT}}(\phi) = -\mathbb{E}_{(x,y)\sim \text{M2D-CORPUS}} \left[\sum_{j=1}^{|y|} \log \pi_\phi(y_j \mid x, y_{<j}) \right]. \tag{2}$$

Only tokens in the target response y contribute to the loss.

For dialogue examples, the expected loss is the cross-entropy between $\bar{q}(\cdot \mid x)$ and $\pi_\phi(\cdot \mid x)$, averaged over inputs. An unrestricted policy minimizes this expectation by matching these conditional distributions. State-informed targets therefore supervise how evidence about the user should affect a response, while averaging over states that x cannot distinguish. The objective transfers informed response behavior without requiring explicit state reconstruction.

4 Training Human-Aware Language Models

We develop M2D-CHAT by training language models on the supervision generated by M2D-SIM. This section presents the training data and recipe, validates the simulator's ability to maintain personal context, and evaluates the resulting models on personalization and mental-state reasoning.

4.1 Training Data and Recipe

Running M2D-SIM over 289 personas yields M2D-CORPUS, including a subset of 6,330 multi-turn conversations analyzed here. We characterize scenario coverage, persona specializations, and user behavior. Scenario categories are recorded during generation; the behavioral analysis classifies generated user messages by embedding-based matching to mode descriptions (Section F.4).

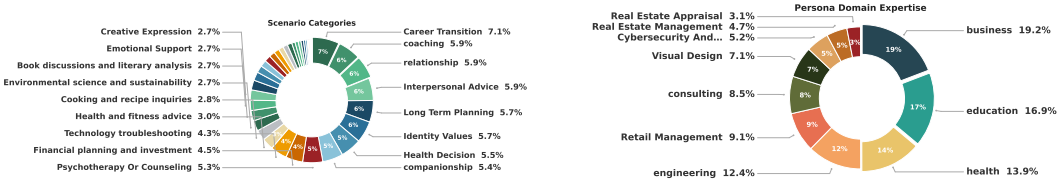

Figure 3: Scenario and persona coverage. The 6,330-conversation analysis subset spans 47 scenario categories and 10 persona-specialization clusters. **(a)** Scenario shares across affective, high-frequency, and lifelong conversations; labels mark categories with shares of at least 2.7%. **(b)** Specializations grouped by embedding similarity. Both panels report shares of conversations in the analysis subset.

Scenarios. Every conversation carries a scenario category assigned by its constructor. The subset spans 47 categories in the three families introduced above: affective (1,428 conversations), high-frequency (2,020), and lifelong (2,882). The coverage is long-tailed rather than concentrated on a few topics: the top 9, 18 and 28 categories represent ∼52%, 81% and 96% of the data (Figure 3a).

Personas. Each persona declares a free-form `specialization`; the pool contains 153 distinct values, which we cluster into 10 groups for visualization (Figure 3b). The personas are grounded in five focal countries, and the broader rollout corpus extends to a global pool.

The behavioral analysis measures the content of generated user turns independently of the controller's intended mode. Figure 4 reports counts for the 14 displayed modes across six families, with each mode decomposed by conversation source. Section H includes the two fallback modes.

The generation pipeline supports additional personas, scenario categories, and behavioral modes without human annotation of each generated dialogue. Section F reports detailed corpus statistics.

Training mixture. The SFT mixture contains 8,244 examples, comprising 3,312 dialogue examples and 4,932 QA examples for persona memory and preference generation or classification (Table 8).

Training. Our primary backbone is Qwen2.5-7B-Instruct [40]. To test whether data scaling depends on the backbone, we also train Llama-3.1-8B-Instruct [13] and OLMo-3-7B-Instruct [35]. We fine-tune each open-source backbone using four fractions of the SFT mixture: 1/8, 1/4, 1/2, and ALL. We keep the training recipe fixed across model families and data scales. Training uses LoRA [16] with 4-bit quantization [7], response-only masking, and AdamW [27] with a cosine learning-rate schedule.

4.2 Pilot Study of Simulation Quality

The pilot compares whether M2D-SIM and a **Vanilla** simulator sustain personal context over an interaction. Both conditions use the same backbone and persona, while Vanilla omits structured state maintenance and behavior guidance. This comparison evaluates the two components together. Section D.2 reports dimension-level judge scores and a separate component ablation.

The pilot shows larger differences favoring M2D-SIM at later stages of interaction (Figure 5). Panel **A** marks the initial difference in user persona specificity as nonsignificant and reports d=0.66 at turn 20. Topic-depth differences increase from d=0.16 over the first five turns to d=1.22 over the first 30 turns, then reach d=1.02 over all 40 turns (**E**). Panel **D** separately measures assistant personalization and reports a final score gap of 0.51. The effect-size trajectories in **C** and **F** have positive correlations with turn index (r=0.90 and r=0.79), without increasing at every measured turn.

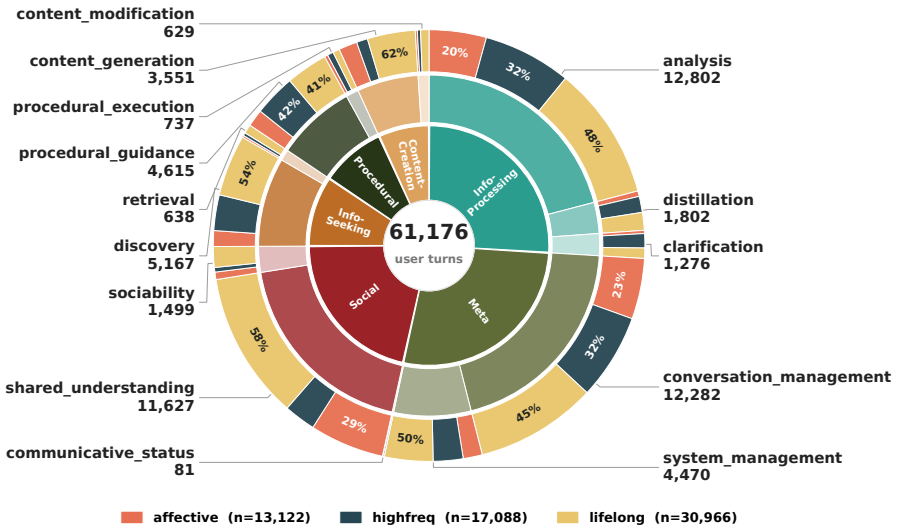

Figure 4: Behavioral coverage of simulated users. The 61,176 plotted turns cover 14 modes across six behavior families. The rings show families, modes, and conversation sources from the center outward. Sector sizes encode turn counts; percentages are source shares within each mode. Mode assignments follow the embedding-based classifier in Section F.4.

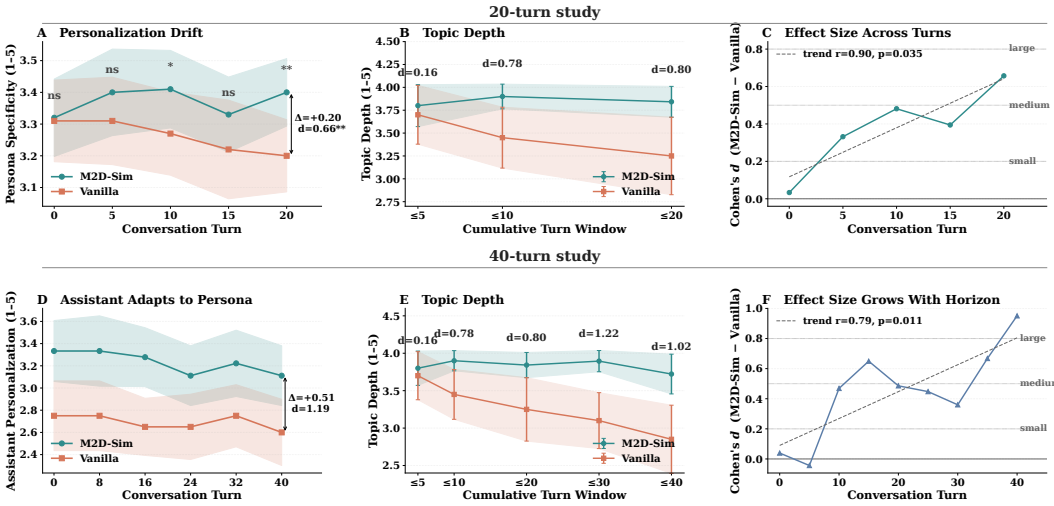

Figure 5: Simulation quality across conversation turns. M2D-SIM shows larger advantages over Vanilla at later turns with backbones and personas held fixed. **A** measures user persona specificity and **D** assistant personalization; **B, E** measure topic depth over cumulative windows. Judge scores range from 1 to 5. **C, F** report Cohen's d, with positive values favoring M2D-SIM; r gives the correlation with turn index. Rows show 20 turns (top) and 40 turns (bottom). Vanilla omits both state maintenance and behavior guidance.

4.3 Evaluation Protocol

Our evaluation separates the quality of simulated supervision from the capabilities learned by the student. Human assessment audits the training conversations, while student evaluation combines personalization with Theory of Mind (ToM). We pair these domains because mental-state attribution and its application to behavior can diverge [14]. Joint gains test whether training improves both the use of personal context in assistance and reasoning about other agents' beliefs and actions.

Human assessment of supervision. A human audit evaluates whether the filtered conversations provide coherent, state-consistent demonstrations. Annotators assessed a uniform sample of 1,240 retained conversations using a binary rubric covering persona consistency, trajectory coherence, state-response consistency, and response relevance. Of these, 1,216 passed (98.06%), supporting the quality of the supervision supplied to the student (Section F.6).

Personal context and preference use. The personalization suite tests whether models apply conversational context to a user's subsequent request. **PersonaMem-v1** tests memory and adaptation to changing profiles [18]; **PersonaMem-v2** emphasizes implicit preferences in task-oriented dialogue [19]. **PrefEval** tests adherence to preferences conveyed explicitly or indirectly in earlier conversation [57]. We report answer-selection and generation scores separately to distinguish recognizing a suitable response from producing one, using the task and backbone coverage in Table 5.

Belief and action reasoning. The ToM suite tests whether models track another agent's knowledge and its consequences for action. **ToMi** evaluates first- and second-order beliefs in narratives with unequal access to information [23]. **BigToM** links percepts, beliefs, desires, and actions through a causal scenario structure [10]. We report its Forward Belief and Forward Action tasks to assess both state attribution and behavior prediction.

Comparison models. We compare M2D-CHAT with the unmodified backbone and with task-specific methods implemented on the same backbone. The personalization baselines are PersonaVLM [34], HumanLM [53], LLMoPt [30], and Qwen2.5-7B-Instruct augmented with Mem0 [3]; the ToM baselines are AutoToM [56] and Thought-Tracing [21]. We include GPT-4o-mini [36] and GPT-5-mini [48] as proprietary reference points.

Comparison and reporting protocol. The method comparison uses Qwen2.5-7B-Instruct; the scaling study compares Qwen, Llama, and OLMo with their own unmodified backbones at the training fractions in Section 4.1. We report scores as percentages and gains as percentage-point differences, retaining separate results for each task and backbone. Table 5 provides the full scaling results, and Table 3 summarizes the metrics shared by all three backbones.

4.4 Evaluation Results

Scaling M2D-CORPUS improves personalization across model families. M2D-SIM can expand the training data without human annotation per-dialog. We tested scalability by training three open-source backbones in four fractions of the same M2D-CORPUS mixture. For each base, the full mixture is best in every reported personalization metric (Figure 6 and Table 5). PrefEval-Gen gains most (26.6 to 40.9 points); the two PersonaMem metrics improve by smaller margins across all three backbones. The trajectories differ: Qwen records its largest PrefEval-Gen gain at the final scale, Llama realizes most of its gain by $1/4$, and OLMo improves more evenly. The sweep supports M2D-CORPUS as scalable personalization supervision in the corpus sizes tested.

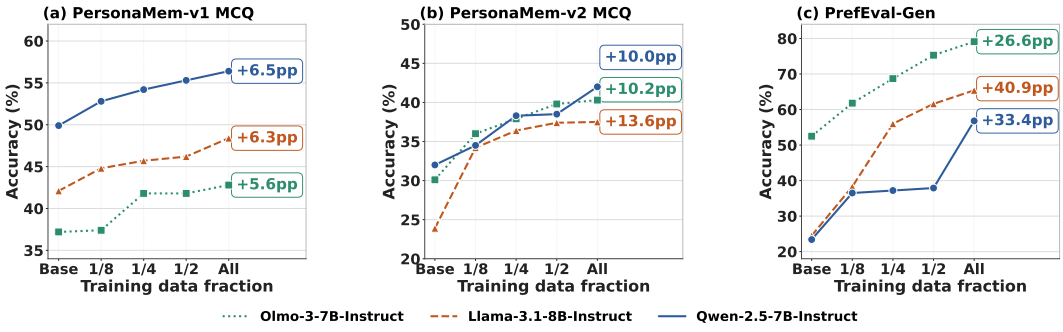

Figure 6: Personalization performance across training scales. The full M2D-CORPUS mixture gives the highest accuracy on each plotted metric for all three backbones. Accuracy is reported on **(a)** PersonaMem-v1 MCQ, **(b)** PersonaMem-v2 MCQ, and **(c)** PrefEval-Gen. Labels report the absolute gain from the unmodified backbone to the full SFT mixture; Table 5 gives the underlying values.

M2D-CHAT outperforms the evaluated personalization baselines. On Qwen2.5-7B-Instruct, M2D-CHAT achieves the best non-proprietary score in all four columns of Table 1. The clearest separation is on the PrefEval generation: its score of 56.8 is 13.2 points above HumanLM, the

Table 1: Personalization benchmark results. M2D-CHAT leads the non-proprietary comparisons on all four metrics. All non-proprietary methods use Qwen2.5-7B-Instruct; entries are accuracy (%). **Bold** marks the best non-proprietary result in each column.

Model / Method	PersonaMem-v1 MCQ	PersonaMem-v2 MCQ	PrefEval Generation	PrefEval Classification
GPT-4o-mini	48.6	37.3	21.1	84.6
GPT-5-mini	61.5	49.2	90.8	99.3
Qwen2.5-7B-Instruct	49.9	32.0	23.4	62.0
PersonaVLM	33.9	24.3	29.8	54.7
HumanLM	38.1	27.3	43.6	78.5
LLMoPt	35.3	22.2	41.5	75.1
Mem0	54.3	39.8	–	–
M2D-CHAT	**56.4**	**42.0**	**56.8**	**78.9**

strongest task-specific baseline in this split. In PersonaMem-v1 and PersonaMem-v2, memory augmentation through Mem0 improves the base model but remains 2.1 and 2.2 points below M2D-CHAT, respectively. The PrefEval classification is closer: M2D-CHAT reaches 78.9, compared to 78.5 for HumanLM, so the 0.4-point margin does not support a strong separation claim. M2D-CHAT exceeds GPT-4o-mini in three of the four evaluations, but remains below GPT-5-mini in all four. These results show that M2D-CORPUS improves a fixed backbone beyond the selected task-specific and memory baselines.

Improvements cover both response generation and answer selection. Table 4 brings two additional evaluation curves into the main comparison, Qwen's PrefEval classification and Llama's PersonaMem-v2 generation, alongside their corresponding generation and selection tasks. On Qwen, PrefEval classification increases from 62.0 to 78.9, alongside the gain in preference-following generation. On Llama, PersonaMem-v2 generation increases from 33.9 to 45.6, while multiple-choice accuracy rises from 23.9 to 37.5. The gains therefore extend across both selecting answers consistent with user information and generating responses that use that information.

The two response formats respond differently to training-data scale. For Qwen, increasing the mixture from one quarter to one half raises PrefEval classification by 8.5 points but generation by only 0.7 points. The final increase to the full mixture then raises generation by 18.9 points and classification by 3.1 points. Llama's PersonaMem-v2 generation follows a different trajectory, reaching 44.0 with one eighth of the data and 45.6 with the full mixture. Endpoint gains alone would conceal these differences in how the models benefit from additional supervision.

M2D-CORPUS improves belief and action prediction on Qwen and Llama. In Qwen2.5-7B-Instruct, M2D-CHAT improves Forward Belief from 31.0 to 44.0 (+13.0) and Forward Action from 23.5 to 31.0 (+7.5). Both gains exceed those of AutoToM (+9.3 and +5.8) and Thought-Tracing (+4.0 and +3.0). On ToMi, M2D-CHAT gains +1.8, compared with +5.0 for AutoToM and +4.4 for Thought-Tracing. Transfer also scales with data on Llama-3.1-8B-Instruct (Figure 7). From Base to ALL, BigToM Forward Belief rises from 18.5 to 43.3 (+24.8), Forward Action from 39.5 to 57.3 (+17.8), and ToMi from 63.6 to 70.6 (+7.0).

Table 2: Belief and action reasoning results. M2D-CHAT leads the Qwen2.5-7B-Instruct comparisons on BigToM, while AutoToM leads on ToMi. Scores are accuracy (%); Belief/Action denote forward prediction. **Bold** marks the best non-proprietary result in each column.

Model / Method	ToMi	Belief	Action
GPT-4o-mini	77.8	52.0	78.0
GPT-5-mini	89.5	93.5	88.0
Qwen2.5-7B-Instruct	80.5	31.0	23.5
+ AutoToM	**85.5**	40.3	29.3
+ Thought-Trace	84.9	35.0	26.5
M2D-CHAT	82.3	**44.0**	**31.0**

Table 3: Training gains across model families. Personalization improves on every shared metric, while ToM gains depend on the backbone. Entries are full-mixture gains over the corresponding Instruct model in percentage points. PM denotes PersonaMem. Complete scores and training scales appear in Table 5.

Backbone	Personalization			Theory of Mind		
	PrefEval Generation	**PM-v1** MCQ	**PM-v2** MCQ	**ToMi**	**BigToM** Forward Belief	**BigToM** Forward Action
Qwen2.5-7B	+33.4	+6.5	+10.0	+1.8	+13.0	+7.5
Llama-3.1-8B	+40.9	+6.3	+13.6	+7.0	+24.8	+17.8
OLMo-3-7B	+26.6	+5.6	+10.2	+2.2	−8.2	−7.0

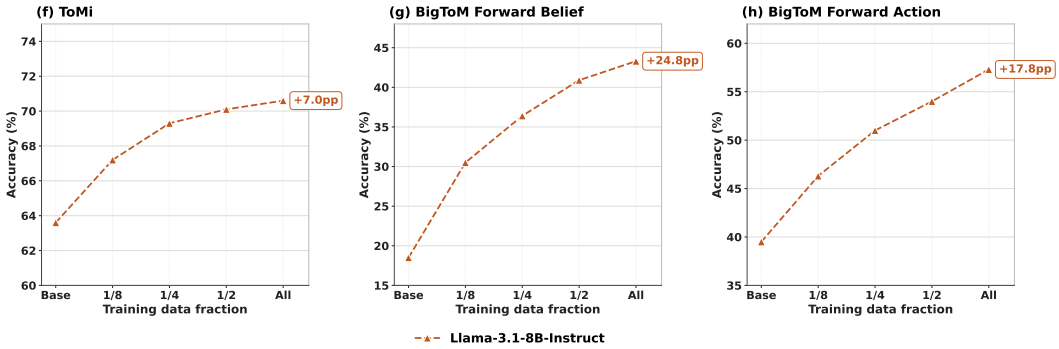

Figure 7: Belief and action prediction across training scales. Llama-3.1-8B-Instruct improves on all three ToM tasks after training on M2D-CORPUS. Panels **(f)**, **(g)**, and **(h)** report accuracy on ToMi, BigToM Forward Belief, and BigToM Forward Action. Base is the unmodified backbone; fractions and All denote the amount of training data. Endpoint labels give gains in percentage points. Table 5 gives scores for all backbones.

Table 4: Response generation and answer selection. Both formats improve along different scaling trajectories. Entries report accuracy (%) at each M2D-CORPUS scale. Base is unmodified; Table 5 gives full results.

Backbone	Benchmark	Task	Base	1/8	1/4	1/2	All
Qwen2.5-7B	PrefEval	Generation	23.4	36.5	37.2	37.9	56.8
		Classification	62.0	63.5	67.3	75.8	78.9
Llama-3.1-8B	PersonaMem-v2	Generation	33.9	44.0	45.3	45.3	45.6
		MCQ	23.9	34.2	36.4	37.4	37.5

Llama improves across both domains with one quarter of the data. At the 1/4 scale (Table 5), Llama reaches 56.0 on PrefEval generation, 36.4 on BigToM Forward Belief, and 51.0 on BigToM Forward Action. These are gains of 31.5, 17.9, and 11.5 percentage points over the base model. At the same scale, ToMi improves from 63.6 to 69.3, and both PersonaMem-v2 response formats improve (Table 4). Increasing the mixture to full scale yields further gains on all these tasks, including another 9.4 points on PrefEval generation and 6.9 points on Forward Belief. The joint improvement is therefore present at an intermediate data scale and continues as supervision increases.

Joint evaluation distinguishes broad gains from task-specific effects. The backbone comparison reveals why personalization alone would give an incomplete account of human-aware learning (Table 3). OLMo gains 26.6 points on preference-following generation and improves on both PersonaMem benchmarks and ToMi, yet loses 8.2 and 7.0 points on BigToM Forward Belief and Forward Action. Qwen and Llama show gains in both domains, with Llama recording the largest improvements in belief and action prediction among the three backbones. The joint evaluation establishes benefits beyond personalized responses for two model families and identifies a concrete limit to that generality in the third. These findings make reasoning about people an empirical criterion for assistant training alongside the quality of the assistance itself.

5 Conclusion

Mind2Dialogue makes the mental states that generate user behavior available for constructing assistant supervision. Shared-state user simulation gives the Oracle information about why a user acts, allowing it to demonstrate assistance informed by beliefs, goals, and circumstances that remain partly implicit in conversation. M2D-SIM, M2D-CORPUS, and M2D-CHAT implement this principle through controlled simulation, corpus construction, and distillation with the state withheld from the student.

Training improves every reported personalization metric across three model families. Qwen and Llama also improve on belief and action prediction, while OLMo's mixed results show why assistance and mental-state reasoning must be assessed together. The broader research opportunity is to determine which experiences teach models to sustain useful assistance as people's knowledge, goals, and circumstances change. Human-aware learning makes that capacity to support a person's understanding and decisions an explicit objective of language-model training.

References

[1] Chris L. Baker, Julian Jara-Ettinger, Rebecca Saxe, and Joshua B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. *Nature Human Behaviour*, 1:0064, 2017. doi: 10.1038/s41562-017-0064. URL https://doi.org/10.1038/s41562-017-0064.

[2] Zhuang Chen, Jincenzi Wu, Jinfeng Zhou, Bosi Wen, Guanqun Bi, Gongyao Jiang, Yaru Cao, Mengting Hu, Yunghwei Lai, Zexuan Xiong, et al. Tombench: Benchmarking theory of mind in large language models. *arXiv preprint arXiv:2402.15052*, 2024.

[3] 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.

[4] Herbert H. Clark and Susan E. Brennan. Grounding in communication. In Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley, editors, *Perspectives on Socially Shared Cognition*. American Psychological Association, 1991. doi: 10.1037/10096-006. URL https://doi.org/10.1037/10096-006.

[5] Katherine M. Collins, Ilia Sucholutsky, Umang Bhatt, Kartik Chandra, Lionel Wong, Mina Lee, Cedegao E. Zhang, Tan Zhi-Xuan, Mark Ho, Vikash Mansinghka, Adrian Weller, Joshua B. Tenenbaum, and Thomas L. Griffiths. Building machines that learn and think with people. *Nature Human Behaviour*, 8, 2024. doi: 10.1038/s41562-024-01991-9. URL https://doi.org/10.1038/s41562-024-01991-9.

[6] Sam Davidson, Salvatore Romeo, Raphael Shu, James Gung, Arshit Gupta, Saab Mansour, and Yi Zhang. User simulation with large language models for evaluating task-oriented dialogue. *arXiv preprint arXiv:2309.13233*, 2023.

[7] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115, 2023.

[8] Feiyu Duan, Xuanjing Huang, and Zhongyu Wei. LifeSim: Long-horizon user life simulator for personalized assistant evaluation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 20419–20463, 2026. doi: 10.18653/v1/2026.findings-acl.1022. URL https://aclanthology.org/2026.findings-acl.1022/.

[9] David C. Funder. Towards a resolution of the personality triad: Persons, situations, and behaviors. *Journal of Research in Personality*, 40(1):21–34, 2006.

[10] Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. Understanding social reasoning in language models with language models. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

[11] Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*, 2024.

[12] Noah D. Goodman and Michael C. Frank. Pragmatic language interpretation as probabilistic inference. *Trends in Cognitive Sciences*, 20(11), 2016. doi: 10.1016/j.tics.2016.08.005. URL https://doi.org/10.1016/j.tics.2016.08.005.

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

[14] Yuling Gu, Oyvind Tafjord, Hyunwoo Kim, Jared Moore, Ronan Le Bras, Peter Clark, and Yejin Choi. SimpleToM: Exposing the gap between explicit ToM inference and implicit ToM application in LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=iE2JmbRJow.

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NeurIPS Deep Learning Workshop*, 2015.

[16] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1 (2):3, 2022.

[17] Pegah Jandaghi, XiangHai Sheng, Xinyi Bai, Jay Pujara, and Hakim Sidahmed. Faithful persona-based conversational dataset generation with large language models. In *Proceedings of the 6th Workshop on NLP for Conversational AI (NLP4ConvAI 2024)*, pages 114–139, 2024.

[18] Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J Taylor, and Dan Roth. Know me, respond to me: Benchmarking llms for dynamic user profiling and personalized responses at scale. *arXiv preprint arXiv:2504.14225*, 2025.

[19] Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, et al. Personamem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025.

[20] Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Le Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. FANToM: A benchmark for stress-testing machine theory of mind in interactions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 14397–14413, 2023. doi: 10.18653/v1/2023.emnlp-main.890.

[21] Hyunwoo Kim, Melanie Sclar, Tan Zhi-Xuan, Lance Ying, Sydney Levine, Yang Liu, Joshua B Tenenbaum, and Yejin Choi. Hypothesis-driven theory-of-mind reasoning for large language models. *arXiv preprint arXiv:2502.11881*, 2025.

[22] Jiho Kim, Junseong Choi, Woosog Chay, Daeun Kyung, Yeonsu Kwon, Yohan Jo, and Edward Choi. ProPerSim: Developing proactive and personalized AI assistants through user-assistant simulation. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

[23] Matthew Le, Y-Lan Boureau, and Maximilian Nickel. Revisiting the evaluation of theory of mind through question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5872–5877, 2019.

[24] Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. Hello again! LLM-powered personalized agent for long-term dialogue. *arXiv preprint arXiv:2406.05925*, 2024.

[25] Shuyue Stella Li, Bhargavi Paranjape, Kerem Oktar, Zhongyao Ma, Gelin Zhou, Lin Guan, Na Zhang, Sem Park, Lin Chen, Diyi Yang, Yulia Tsvetkov, and Asli Celikyilmaz. HorizonBench: Long-horizon personalization with evolving preferences, 2026. URL https://arxiv.org/abs/2604.17283.

[26] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations (ICLR)*, 2016.

[27] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[28] Han Luo and Guy Laban. Spasm: Stable persona-driven agent simulation for multi-turn dialogue generation, 2026. URL https://arxiv.org/abs/2604.09212.

[29] Jiani Luo, Xiaoyan Zhao, Yang Zhang, Shuyi Miao, Bingbing Xu, Stefan Konigorski, and Tat-Seng Chua. Know you before you speak: User-state modeling for LLM personalization in multi-turn conversation, 2026. URL https://arxiv.org/abs/2605.24647.

[30] Yuchen Ma, Yue Huang, Wenjie Wang, Xiaonan Luo, Xiangliang Zhang, and Stefan Feuerriegel. Synthetic interaction data for scalable personalization in large language models. *arXiv preprint arXiv:2602.12394*, 2026.

[31] Lucie Charlotte Magister, Katherine Metcalf, Yizhe Zhang, and Maartje ter Hoeve. On the way to LLM personalization: Learning to remember user conversations. *arXiv preprint arXiv:2411.13405*, 2024.

[32] Walter Mischel and Yuichi Shoda. A cognitive-affective system theory of personality: Reconceptualizing situations, dispositions, dynamics, and invariance in personality structure. *Psychological Review*, 102(2):246–268, 1995. doi: 10.1037/0033-295X.102.2.246.

[33] Tarek Naous, Philippe Laban, Wei Xu, and Jennifer Neville. Flipping the dialogue: Training and evaluating user language models. *arXiv preprint arXiv:2510.06552*, 2025.

[34] Chang Nie, Chaoyou Fu, Yifan Zhang, Haihua Yang, and Caifeng Shan. Personavlm: Long-term personalized multimodal llms. *arXiv preprint arXiv:2604.13074*, 2026.

[35] Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

[36] OpenAI. Gpt-4o mini: Advancing cost-efficient intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, 2024. Accessed: 2026-05-07.

[37] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023. URL https://arxiv.org/abs/2304.03442.

[38] Emiliano Penaloza, Dheeraj Vattikonda, Nicolas Gontier, Alexandre Lacoste, Laurent Charlin, and Massimo Caccia. Privileged information distillation for language models, 2026. URL https://arxiv.org/abs/2602.04942. ICML 2026.

[39] Cheng Qian, Zuxin Liu, Akshara Prabhakar, Jielin Qiu, Zhiwei Liu, Haolin Chen, Shirley Kokane, Heng Ji, Weiran Yao, Shelby Heinecke, Silvio Savarese, Caiming Xiong, and Huan Wang. UserRL: Training interactive user-centric agent via reinforcement learning. *arXiv preprint arXiv:2509.19736*, 2025. URL https://arxiv.org/abs/2509.19736.

[40] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

[41] Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, S. M. Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4218–4227, 2018.

[42] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, 2019. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

[43] Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. Lamp: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7370–7392, 2024.

[44] Ivan Sekulić, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, Andre Ferreira Manso, and Roland Mathis. Reliable llm-based user simulator for task-oriented dialogue systems. *arXiv preprint arXiv:2402.13374*, 2024.

[45] Natalie Shapira, Mosh Levy, Seyed Hossein Alavi, Xuhui Zhou, Yejin Choi, Yoav Goldberg, Maarten Sap, and Vered Shwartz. Clever hans or neural theory of mind? Stress testing social reasoning in large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2024.

[46] Renee Shelby, Fernando Diaz, and Vinodkumar Prabhakaran. Taxonomy of user needs and actions, 2025. URL https://arxiv.org/abs/2510.06124.

[47] Kazutoshi Shinoda, Nobukatsu Hojo, Kyosuke Nishida, Saki Mizuno, Keita Suzuki, Ryo Masumura, Hiroaki Sugiyama, and Kuniko Saito. ToMATO: Verbalizing the mental states of role-playing LLMs for benchmarking Theory of Mind. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1520–1528, 2025. doi: 10.1609/aaai.v39i2.32143.

[48] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

[49] Tomer Ullman. Large language models fail on trivial alterations to theory-of-mind tasks. *arXiv preprint arXiv:2302.08399*, 2023.

[50] Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009.

[51] Ruiyi Wang, Haofei Yu, Wenxin Zhang, Zhengyang Qi, Maarten Sap, Graham Neubig, Yonatan Bisk, and Hao Zhu. SOTOPIA-π: Interactive learning of socially intelligent language agents. *arXiv preprint arXiv:2403.08715*, 2024.

[52] Zhen Wang, Yufan Zhou, Zhongyan Luo, Lyumanshan Ye, Adam Wood, Man Yao, Saab Mansour, and Luoshang Pan. Deeppersona: A generative engine for scaling deep synthetic personas. In *NeurIPS 2025 Workshop on Bridging Language, Agent, and World Models for Reasoning and Planning*, 2025. URL https://arxiv.org/abs/2511.07338.

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

[54] Hainiu Xu, Runcong Zhao, Lixing Zhu, Jinhua Du, and Yulan He. OpenToM: A comprehensive benchmark for evaluating theory-of-mind reasoning capabilities of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 8593–8623, 2024. doi: 10.18653/v1/2024.acl-long.466.

[55] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2204–2213, 2018. doi: 10.18653/v1/P18-1205.

[56] Zhining Zhang, Chuanyang Jin, Mung Yao Jia, Shunchi Zhang, and Tianmin Shu. Autotom: Scaling model-based mental inference via automated agent modeling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[57] Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. Do LLMs recognize your preferences? evaluating personalized preference following in LLMs. *arXiv preprint arXiv:2502.09597*, 2025.

[58] Yue Zhao, Xiaoyu Wang, Dan Wang, Zhonglin Jiang, Qingqing Gu, Teng Chen, Ningyuan Xi, Jinxian Qu, Yong Chen, and Luo Ji. Dream to chat: Model-based reinforcement learning on dialogues with user belief modeling. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 4764–4781, 2025. doi: 10.18653/v1/2025.findings-emnlp.256. URL https://aclanthology.org/2025.findings-emnlp.256/.

[59] Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. MemoryBank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[60] Xuhui Zhou, Weiwei Sun, Weihua Du, Jiarui Liu, Haojia Sun, Qianou Ma, Tongshuang Wu, Yiming Yang, and Maarten Sap. OdysSim: Building foundation models for human behavior simulation, 2026. URL https://arxiv.org/abs/2606.14199.

A Limitations

Mind2Dialogue studies synthetic supervision for a setting in which real long-horizon dialogue is difficult to collect. We identify four limitations of the present evaluation.

Dependence on Oracle quality. All data-generation experiments use GPT-4o-mini as both the user simulator and the Oracle assistant. The corpus can therefore inherit this model's errors, stylistic biases, and limitations in representing user states. We do not test whether a stronger teacher, an ensemble, or a different simulator family would improve the student.

Backbone-dependent Theory-of-Mind transfer. Qwen2.5-7B and Llama-3.1-8B improve on all three measured ToM tasks. OLMo-3-7B improves on ToMi but declines on both BigToM tasks. For OLMo, the personalization gains persist (PrefEval-Gen $+26.6$, PersonaMem-v2 $+10.2$), while BigToM regresses (Forward Belief -8.2, Forward Action -7.0). Differences in pretraining, instruction tuning, optimization, or evaluation sensitivity could contribute to this result; the present experiments do not distinguish among them. Theory-of-Mind transfer should therefore be treated as an empirical observation for two backbones rather than a guaranteed property of the method.

Latent-state schema. The state document maintained by M2D-SIM uses a hand-designed schema containing conversational context, stable attributes, transient attributes, and behavioral mode. This representation may omit relevant aspects of users or encode distinctions that do not transfer to real interactions. We do not compare alternative schemas or test sensitivity to individual state fields. Extending the schema may also change simulation dynamics and data quality, so compatibility with the remaining pipeline requires empirical validation.

Validation against real long-horizon dialogue. We do not validate the simulated trajectories or trained models against real long-horizon human–assistant interactions. Instead, evaluation relies on public personalization and Theory-of-Mind benchmarks (PersonaMem, PrefEval, BigToM, and ToMi) and on LLM-based judgments of synthetic trajectories. These evaluations may not capture whether users find the responses accurate, helpful, or appropriately calibrated over time. Longitudinal studies with informed consent and appropriate privacy protections are needed before drawing conclusions about real-user utility.

B Broader Impact

Positive impact. Personalization research often relies on private interaction logs or on synthetic conversations generated from static persona descriptions. Mind2Dialogue provides an alternative source of training data in which both the simulated user and assistant are conditioned on a constructed state. The pipeline does not require collecting private human–assistant conversation histories, which reduces one source of privacy risk. Because the state and generation metadata are available to the data producer, the corpus can also support controlled analyses of persona, scenario, and behavioral coverage. A public release can facilitate replication and comparison, although synthetic generation does not by itself guarantee demographic balance, factual accuracy, or privacy safety.

Potential negative impact and mitigation. Systems designed to infer user goals, beliefs, or affect may also enable targeted persuasion, dependency, or unwanted profiling. Synthetic personas and state descriptions can encode stereotypes, and a model may express unwarranted confidence about mental states that are inherently uncertain. We do not evaluate manipulation, demographic bias, privacy leakage, or safety in mental-health-adjacent settings. Open release supports auditing but also broadens access to the capability. Any downstream deployment should therefore evaluate these risks directly, communicate uncertainty, provide user control over personalization and memory, and apply domain-appropriate safety and data governance measures. We release M2D-CORPUS as a research artifact rather than a production-ready dataset.

C Extended Related Work

We organize the literature around four requirements for human-aware assistance, namely access to personal context, realistic interaction, reasoning about people, and supervision that remains useful when user states are unobserved. The comparison below identifies what each research direction contributes and where shared-state supervision changes the learning problem.

C.1 Personalization and User Modeling

Recent benchmarks emphasize implicit preferences that are inferred from user behavior [57, 19, 25]. PersonaMem-v2 additionally studies how preference supervision can support reinforcement fine-tuning and agentic memory learning [19]. LaMP evaluates personalized classification and generation and studies retrieval from user profiles [43]. MemoryBank updates and retrieves memories from prior interactions, Mem0 extracts and consolidates salient conversational information, and long-term dialogue agents combine memory with user modeling [59, 3, 24]. PLUM takes a parameter-based approach, augmenting previous conversations into QA examples for user-specific adapter training [31]. These approaches determine how personal information is retained and made available to the model. Mind2Dialogue constructs demonstrations of how personal context informs assistance.

Explicit user-state models extend personalization to reasoning about changes that are only partly observable in dialogue. PUMA formulates interaction as decision-making under partial observability, maintains beliefs over user states, and selects actions using predicted state transitions [29]. DreamCUB predicts future utterances and user beliefs in a dialogue world model and uses model-based reinforcement learning to improve the dialogue policy [58]. Both methods make user-state reasoning part of assistant decision-making. Mind2Dialogue supplies a complementary source of supervision by constructing states during simulation and exposing them to the Oracle before generating training responses. The student learns those responses with the evolving state withheld.

C.2 Synthetic Dialogue and User Simulation

Persona-conditioned dialogue commonly starts from a fixed profile [55]. Synthetic-Persona-Chat uses a generator and critics to expand persona-based conversations while checking their quality [17]. Persona Hub broadens the population available for data synthesis, while DeepPersona increases the depth and internal detail of synthetic profiles [11, 52]. Profile diversity and dialogue quality control are useful foundations, but a profile alone does not specify the sequence of mental states through which a user responds to an unfolding interaction. M2D-SIM adds an explicit state trajectory and uses each updated state to guide both user behavior and Oracle assistance.

Interactive simulation models how users pursue goals and respond to events over time. Task-oriented user simulators support controlled dialogue evaluation [6, 44], and UserLM trains the user role directly on human conversations conditioned on high-level intent [33]. Generative Agents combine memory, reflection, and planning to produce behavior over time [37]. LifeSim models long-horizon user lives and evolving intentions for personalized assistant evaluation [8]. HumanLM aligns latent states with real user responses, and OdysSim develops models trained across a broad collection of human behavior tasks [53, 60]. These systems demonstrate that dynamic state and realistic behavior are established ingredients of simulation. We focus on *who observes the simulator-defined user state when assistant responses are generated.* In M2D-SIM, the user simulator updates a structured state, and an Oracle assistant directly observes that same state when producing responses.

Training through simulated interaction also requires deciding which aspect of the assistant's behavior receives supervision. PersonaGym generates dynamic preference interactions, while its associated PPOpt method learns to rewrite prompts from inferred user profiles [30]. ProPerSim adapts a proactive assistant from user-specific feedback within a simulation of daily activities [22]. Such feedback can improve assistants while preserving the separation between the user's internal state and the assistant's current estimate of it. Our Oracle receives the simulator-defined state directly when constructing response targets. This choice provides an explicit source of information for the demonstration, addressing the gap between generating a realistic user and generating an assistant response informed by that user.

C.3 Social Intelligence and Mental-State Reasoning

Human social cognition provides a basis for treating people's beliefs and goals as part of the problem an assistant must solve [1, 5]. Computational work on social intelligence examines both successful interaction and the reasoning needed to interpret other agents. SOTOPIA-π develops social behavior through behavior cloning and self-reinforcement on evaluated interactions [51]. Its learning signal concerns the quality of the interaction; Mind2Dialogue additionally specifies the user state available to the assistant that generates supervision. This distinction concerns how demonstrations are constructed and can be combined with improvements in interaction-based learning.

Theory-of-Mind (ToM) benchmarks test whether models track beliefs and other mental states in narratives and conversations [23, 10, 20, 2, 54]. FANToM makes information asymmetry central to its conversational evaluation, while OpenToM and ToMBench broaden the range of characters, states, and questions. ToMATO couples characterized speakers with verbalized mental states [47]. Its agents' thoughts, goals, and personalities are hidden from their partners, and the thoughts supply mental-state QA labels. ToMATO also studies fine-tuning on separately generated QA data. Mind2Dialogue exposes the user's evolving state to an Oracle that generates assistant responses, which then supervise a student without state access.

Joint evaluation on personalization and ToM tests two consequences of training models to attend to people. Personalization measures whether the assistant uses information about the user when responding; ToM measures whether the same training benefits explicit reasoning about another agent's beliefs and actions. The two suites provide separate behavioral observations, and sensitivity to perturbations remains relevant when interpreting ToM scores [49, 45]. The mixed BigToM results across backbones further delimit the empirical connection reported in this paper.

C.4 Learning with Privileged Information

Learning with privileged information allows training to benefit from information unavailable at test time [50]. Knowledge distillation transfers a teacher's predictions, and generalized distillation connects this principle to teachers and students receiving different representations of an example [15, 26]. This literature motivates our use of a state-informed Oracle to supervise an assistant that cannot directly observe the evolving user state.

Recent work studies how to optimize this transfer for language models. Privileged Information Distillation jointly trains conditioned teachers and students with shared parameters, and its on-policy variant uses a conditioned teacher to regularize student behavior [38]. By contrast, our Oracle is frozen; the teacher and student use separate parameters; and transfer is supervised. Shared-state simulation supplies the additional information and links it to user behavior before distillation begins. The resulting contribution is a construction of supervision for human-aware assistance that can operate with standard language model training.

D Complete Evaluation Results and Simulation Analysis

We report the complete student scaling results underlying Section 4.4, followed by the simulator validation and component ablation supporting Section 4.2.

D.1 Complete Data Scaling Results

Table 5 retains every reported score and training fraction. The main-text summary in Table 3 selects full-mixture gains on metrics available for all three backbones.

Table 5: Complete results across training scales. Full-mixture training improves every reported personalization metric; ToM outcomes vary across backbones. Panels **(a)** and **(b)** report personalization and mental-state reasoning, respectively. All entries are accuracy (%). **Bold** marks the best score within each backbone and metric; parenthesized values report the absolute change from the base model for full-mixture SFT. Base denotes the unmodified model, fractions specify the training-mixture size, and dashes indicate unreported results.

(a) Personalization benchmarks

Backbone	Scale	PrefEval Gen.	PrefEval Cls.	PM-v1 MCQ	PM-v2 MCQ	PM-v2 Gen.
Qwen2.5-7B	Base	23.4	62.0	49.9	32.0	–
	SFT-1/8	36.5	63.5	52.8	34.5	–
	SFT-1/4	37.2	67.3	54.2	38.3	–
	SFT-1/2	37.9	75.8	55.3	38.5	–
	SFT-All	**56.8** (+33.4)	**78.9** (+16.9)	**56.4** (+6.5)	**42.0** (+10.0)	–
Llama-3.1-8B	Base	24.5	–	42.1	23.9	33.9
	SFT-1/8	38.3	–	44.8	34.2	44.0
	SFT-1/4	56.0	–	45.7	36.4	45.3
	SFT-1/2	61.6	–	46.2	37.4	45.3
	SFT-All	**65.4** (+40.9)	–	**48.4** (+6.3)	**37.5** (+13.6)	**45.6** (+11.7)
OLMo-3-7B	Base	52.5	–	37.2	30.1	–
	SFT-1/8	61.8	–	37.4	36.0	–
	SFT-1/4	68.7	–	41.8	37.9	–
	SFT-1/2	75.3	–	41.8	39.8	–
	SFT-All	**79.1** (+26.6)	–	**42.8** (+5.6)	**40.3** (+10.2)	–

(b) Theory-of-Mind transfer benchmarks

Backbone	Scale	ToMi	BigToM Forward Belief	BigToM Forward Action
Qwen2.5-7B	Base	80.5	31.0	23.5
	SFT-1/8	–	36.5	–
	SFT-1/4	–	39.5	–
	SFT-1/2	–	42.0	–
	SFT-All	**82.3** (+1.8)	**44.0** (+13.0)	**31.0** (+7.5)
Llama-3.1-8B	Base	63.6	18.5	39.5
	SFT-1/8	67.2	30.5	46.3
	SFT-1/4	69.3	36.4	51.0
	SFT-1/2	70.1	40.9	54.0
	SFT-All	**70.6** (+7.0)	**43.3** (+24.8)	**57.3** (+17.8)
OLMo-3-7B	Base	78.1	**30.2**	**59.2**
	SFT-1/8	–	–	–
	SFT-1/4	–	–	–
	SFT-1/2	–	–	–
	SFT-All	**80.3** (+2.2)	22.0 (-8.2)	52.2 (-7.0)

D.2 Simulator Validation and Component Ablations

M2D-SIM receives higher trajectory-level judge scores than Vanilla across all five measured dimensions. We evaluate 20 held-out personas using the LLM-as-judge protocol of Luo and Laban [28].

The composite Z-score gap is 1.15 (d=2.12), and M2D-SIM receives the higher aggregate score for all 20 paired personas (binomial $p<0.001$). Table 6 reports all scores and effect sizes.

Table 6: Simulator validation on held-out personas. M2D-SIM scores higher than Vanilla across all five judge dimensions on 20 held-out personas. Vanilla omits state maintenance and behavior guidance. Compare methods within each row because score scales differ; the composite row reports standardized scores. Positive Cohen's d favors M2D-SIM. **Bold** marks the higher score.

Dimension	**M2D-SIM**	**Vanilla**	Cohen's d
Oracle Personalization	**3.28**	2.97	0.87
Information Efficiency	**0.32**	0.24	2.05
Cross-Persona Distinctness	**0.87**	0.83	1.08
Turn Novelty	**0.78**	0.69	1.19
Topic Depth	**3.72**	2.72	1.00
Composite Z-score	$+0.57$	-0.58	**2.12**

State maintenance supports persona specificity. A component ablation varies state maintenance and behavior prompting independently (Table 7). Compared to the full simulator, conditions without the state document have lower persona-specificity scores at turn 20 (*Vanilla*: $d=-0.66$, $p<.01$; *Profile-only Oracle*: $d=-0.49$, $p<.05$). The two conditions decline over the evaluated horizon, whereas the state-maintaining conditions increase. Vanilla user messages are $2.4\times$ longer on average, suggesting that verbosity alone does not explain the observed gap.

Table 7: Ablation of state and behavior controls. The full simulator has the highest persona specificity at turn 20. State and Behavior indicate state maintenance and behavior guidance. Scores track user persona specificity at initialization and turn 20; negative d denotes lower final specificity than M2D-SIM. Persona specificity evaluates the simulated user trajectory and therefore does not directly measure the Oracle's access to user state.

Condition	**Components**		**Persona specificity (1–5)**		d **vs. M2D-SIM**
	State	**Behavior**	t=0	t=20	
Vanilla	✗	✗	3.31	3.20↓	-0.66^{**}
Profile-only Oracle	✗	✓	3.35	3.28↓	-0.49^{*}
Stateful (no behavior controller)	✓	✗	3.26	3.35	-0.15
M2D-SIM	✓	✓	3.32	**3.40**	ref.

Note. Vanilla users are $2.4\times$ longer than M2D-SIM users (97.6 vs. 40.6 words per message), so verbosity cannot explain the gap. Per-dimension effects concentrate on *goal* (d=1.03), rather than *identity* ($d=-0.05$) or *communication* ($d=-0.08$). $^{*}p<.05$; $^{**}p<.01$.

E Simulation Algorithm

Algorithm 1 details M2D-SIM and corpus construction (Sections 3.2 and 3.3).

Algorithm 1 Shared-state corpus generation.

Require: Persona p; Scenario $\mathcal{S}_p$; Behavior Mode Families $\mathcal{F}=\{F_1, \ldots, F_6\}$; Horizon T; Threshold θ_{sim}
Ensure: Corpus $\mathcal{D}$, State Trajectory $s_{1:T}$

Scenario sampling

repeat
  $\sigma \sim \text{Unif}\{\text{LIFELONG}, \text{HIGHFREQ}, \text{AFFECTIVE}\}$
  $x \leftarrow \text{PROPOSESCENARIO}(p, \sigma)$
until x is persona-consistent and abstract, and $\max_{x' \in \mathcal{S}_p} \text{LLM-as-Judge}(x, x') \leq \theta_{\text{sim}}$
$\mathcal{S}_p \leftarrow \mathcal{S}_p \cup \{x\}$; $s_0 \leftarrow \text{INITSTATE}(p, x)$; $H_{<1} \leftarrow \varnothing$; $\mathcal{D} \leftarrow \varnothing$

Rollout

for $t = 1$ **to** T **do**
  $s_t \leftarrow f_{\text{state}}(p, s_{t-1}, H_{<t})$ ▷ unresolved goals, affect, trust
  $F \leftarrow \text{SELECTFAMILY}(\mathcal{F}, \text{coverage}_{<t})$
  $b_t \sim \text{Unif}(F)$; $g_t \leftarrow \text{GUIDANCE}(b_t, t)$
  $m_t \leftarrow f_{\text{user}}(p, s_t, H_{<t}, g_t)$
  $a_t \leftarrow f_{\text{oracle}}(p, s_t, H_{<t}, m_t)$
  $H_{<t+1} \leftarrow H_{<t} \,\|\, (m_t, a_t)$
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{(p, (H_{<t}, m_t), a_t)\}$
end for

Retention

if $H_{<T+1}$ passes the four programmatic and two judge checks **then**
  return $\mathcal{D} \cup \text{MAKEQA}(p, s_{1:T}, H_{<T+1})$
else
  discard the trajectory
end if

F Dataset Statistics

The public release of M2D-CORPUS comprises samples drawn from 289 personas: deep-scenario multi-turn conversations, a broader rollout corpus, and QA-format training items (Section F.5 gives the composition of the subset used for training). This section characterizes the 6,330-conversation deep-scenario subset, whose conversations are paired with full latent-state trajectories. Its personas come from five focal countries (U.S., China, Japan, Germany, and India), whereas the broader rollout corpus additionally uses a global persona pool.

F.1 Scenario coverage

All category statistics below are read directly from the `scenario_category` field assigned at generation time. Section 4.1 gives family-level counts; this section reports them by category.

Figure 8 shows the 25 largest categories (left) and cumulative coverage of all 47 categories (right). Figure 3(a) displays category shares and labels those of at least 2.7%.

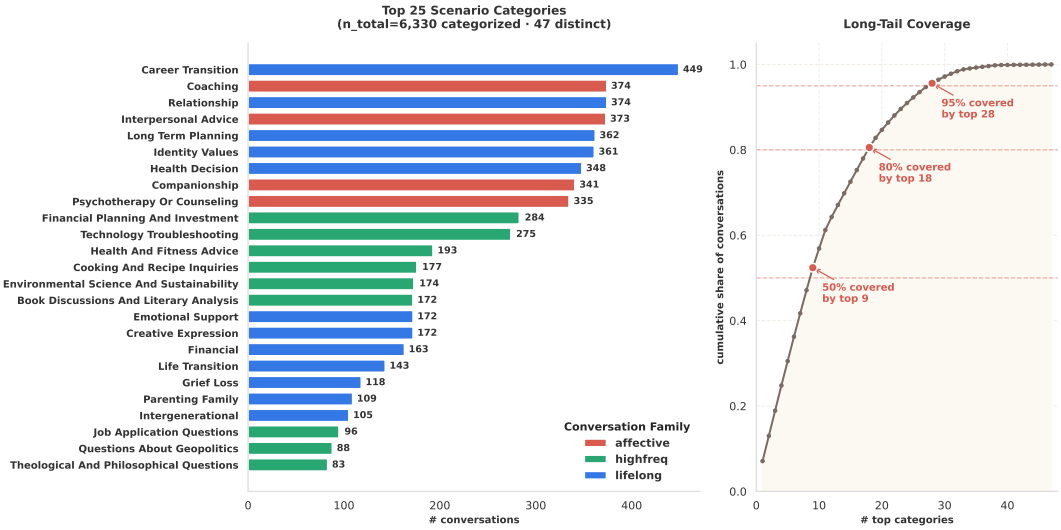

Figure 8: Scenario-category distribution. Scenario frequencies are uneven across the 47 categories in the 6,330-conversation subset. The left panel shows counts for the 25 largest categories, colored by scenario family. The right panel shows cumulative conversation coverage as categories are added in descending frequency.

F.2 Persona specializations

The persona pool contains 153 distinct free-form `specialization` values. We embed these strings with all-MiniLM-L6-v2 [42] and apply K-means with $K = 10$ and random seed 42. Each cluster is labeled by the specialization associated with the most conversations. Figure 3(b) counts one item per conversation and therefore shows the contribution of each specialization cluster to the data.

F.3 Alignment to profile and scenario

For each conversation included in the alignment analysis, we compute the cosine similarity between the assistant text and two references: the static persona and the scenario prompt together with its context. Figure 9(a) and (b) show kernel density estimates of the two distributions for each data source. Panel (c) shows paired values within each source, with the median drawn as a black bar and the mean as a white diamond.

The two references are not interchangeable. Within each analyzed source subset (471 affective, 897 highfreq, and 2,187 lifelong), a paired Wilcoxon signed-rank test finds greater alignment with the scenario than with the static persona ($p < 0.001$ in all three cases).

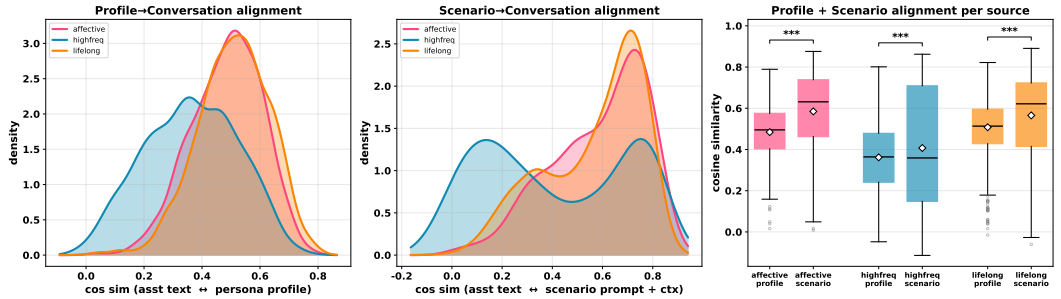

Figure 9: Alignment with personas and scenarios. Assistant text has higher cosine similarity to scenarios than to personas across all three sources. **(a, b)** Similarity distributions for persona and scenario references. **(c)** Paired comparisons, with black bars for medians and white diamonds for means. The subsets contain 471 affective, 897 high-frequency, and 2,187 lifelong conversations; $^{***}p < 0.001$ in Wilcoxon signed-rank tests.

F.4 Behavioral modes

The behavioral analysis classifies the user turns produced by the simulator, because controller selections were not retained as labels in the generated conversations. For each of the 16 TUNA modes, we form a prototype π_m by concatenating the mode description with up to five example user turns from the taxonomy. Let $\phi(\cdot) \in \mathbb{S}^{d-1}$ be the ℓ_2-normalized all-MiniLM-L6-v2 embedding, with $d = 384$. Each turn u receives the label

$$\hat{m}(u) = \arg\max_{m \in \mathcal{M}} \langle \phi(u), \phi(\pi_m) \rangle, \qquad |\mathcal{M}| = 16. \tag{3}$$

The inner product is cosine similarity, so each label identifies the nearest mode prototype.

Figure 4 summarizes 61,176 turns across the 14 displayed modes. The inner ring shows six mode families, the middle ring separates individual modes, and the outer ring shows the contribution of each conversation source within a mode. Figure 10 uses turn positions to describe how family proportions change across the 65,107 user turns in the 6,330-conversation analysis corpus. We normalize each source and turn position after excluding fallback modes.

Formally, a surjection $F : \mathcal{M} \to \mathcal{F} \cup \{\text{Other}\}$ maps the 16 modes to the six functional families $\mathcal{F}$; the two fallback modes (`compound_request` and `default_behavior`) are mapped to Other and excluded before renormalization. Letting $U_{s,t}$ denote the set of user turns at position t from source s, the plotted family proportion is

$$p_{s,t}(f) \;=\; \frac{\big|\{\, u \in U_{s,t} \;:\; F(\hat{m}(u)) = f \,\}\big|}{\big|\{\, u \in U_{s,t} \;:\; F(\hat{m}(u)) \neq \text{Other} \,\}\big|}, \qquad f \in \mathcal{F}, \tag{4}$$

such that $\sum_{f \in \mathcal{F}} p_{s,t}(f) = 1$ for every (s, t). We restrict the analysis to $t \leq 12$ and plot a point only when at least five turns remain after excluding Other for the corresponding source and turn position.

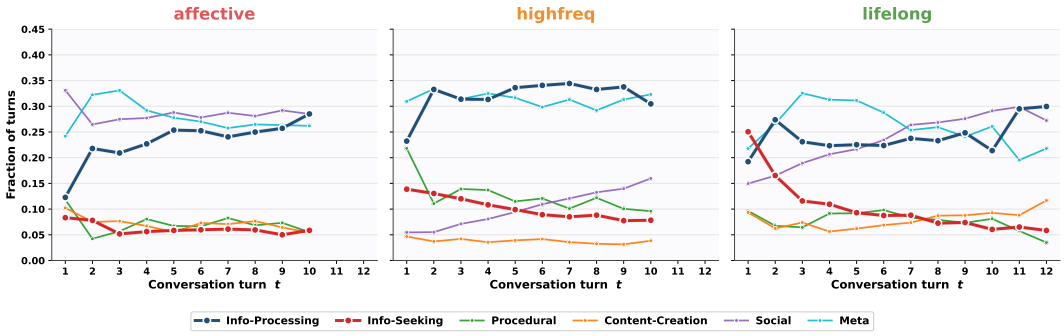

Figure 10: Behavioral composition across dialogue turns. Behavior-family shares vary with turn position and conversation source. Each panel reports family proportions within a source, normalized after excluding fallback modes. Colors identify the six families. Points are shown through turn 12 only when at least five retained turns are available for that source and position.

F.5 SFT mixture composition

The models evaluated in this paper are trained on a subset of the corpus described above. We derive training examples from two views of the same latent-state trajectories. In the dialog view, the student receives the persona and the observable dialog history, with the state-aware Oracle response as the target. In the QA view, the student receives the observable history, a generated question, and answer options when required by the task format. Neither view exposes the latent-state document s_t to the student. This document is available only to the user simulator, Oracle, and the QA item generator.

The QA components provide training examples in the multiple-choice and classification formats used by several evaluations. The dialogue component instead provides free-form targets generated by an Oracle that has access to the latent state. One QA component follows the four-option PersonaMem interface. We manually specify the generation procedure for this component, while M2D-SIM generates each persona, scenario, trajectory, question, and answer option. Training and evaluation data are therefore matched in the interface but disjoint in content (Section 4.3). Performance under this matched interface provides evidence within that setting, but does not by itself establish transfer to formats not represented during training.

Table 8: Composition of the training mixture. The 8,244 examples combine dialogue demonstrations with three QA formats. All components are derived from M2D-SIM trajectories that passed quality control. The mixture contains no instances, histories, questions, or answer options from the evaluation benchmarks.

View	Component	Examples
Dialogue	Multi-turn conversation supervision	3,312
QA	PersonaMem-format MCQ	2,052
QA	Open-ended preference QA	1,442
QA	Preference-classification QA	1,438
Total		8,244

F.6 Quality control

Before inclusion in M2D-CORPUS, each conversation generated by M2D-SIM undergoes six quality checks. Four programmatic checks examine properties that can be determined from the conversation record. Two LLM-judge checks assess whether the simulated user remains consistent with the assigned persona. Section G.5 provides the complete rubrics for both LLM-judge checks.

LLM judge dimensions.

- **Persona consistency**. The judge assigns a Likert score from 1 to 5 based on how consistently the simulated user's messages reflect the assigned profile and behavioral metadata. The judge also provides a written justification.
- **Profile contradiction.** The judge labels each conversation `no_contradiction`, `unclear`, or `contradicts`, according to whether a user turn conflicts with an immutable profile attribute. For a detected conflict, the judge also records the relevant turn index.

Human audit. We conducted a human audit to estimate the quality of conversations retained after automatic filtering. We uniformly sampled 1,240 conversations that had passed all six checks. Annotators evaluated each conversation using a binary rubric that covers persona consistency, trajectory coherence, state-response consistency, and response relevance. Of the sample conversations, 1,216 satisfied the rubric (98.06%). Most of the remaining cases involved prompt seed capture, in which a generic scenario displaced the assigned persona.

G Prompts

G.1 Scenario Generation

G.1.1 Lifelong Scenario

```
You are creating deeply personal conversation-starting messages
for a specific person in a conversation with an AI assistant.
<profile_summary>
{profile_summary}
</profile_summary>
<behavior_metadata>
{behavior_metadata}
</behavior_metadata>
## Narrative Instructions
The full set of scenarios must read as a coherent life story, not isolated
topics. Follow these three rules before selecting categories:
1. Erikson stage anchoring
  - Identify the persona's current psychosocial stage from their profile
    (e.g. Generativity vs. Stagnation for mid-life; Integrity vs.
   Despair for late life).
  - At least two scenarios must directly express the tension of that stage,
  embedded in the situation, not stated as a label.
2. Life phase distribution. Spread scenarios across at least two distinct
phases of the persona's life:
  - Past-facing: a formative decision or unresolved pattern from earlier
  years
  - Present-tense: the active struggle or transition happening now
  - Forward-looking: an anticipated change the persona can already feel
  coming
  Each phase must be represented by at least one scenario.
```

```
3. Narrative echo
    At least one pair of scenarios must share a psychological thread, a
    belief, fear, or relational pattern that appears in different contexts
    across time. Mark the connection in context_note so it is traceable.
<scenario_categories>
Spread scenarios across at least 5 of these categories. Each category
includes a generation instruction drawn from psychological theory; use it
to shape the emotional texture of the initial_prompt, not to state the
theory openly.
- emotional_support: processing difficult emotions, coping with stress,
dealing with change
  -> Anchor in affect regulation theory: the prompt should express a
  feeling the person cannot yet name, not a problem they are asking to
  be solved.
- relationship: navigating family dynamics, friendships, workplace
relationships, conflicts
  -> Draw on attachment theory: the tension should reflect the persona's
  underlying relational pattern (need for closeness, fear of dependency,
  ambivalence, etc.).
- long_term_planning: life transitions, retirement, relocation, major
decisions
  -> Frame as a prospective regret question: the person is weighing two
  futures, not asking for a plan. At least one option must feel like a
  loss.
- career_transition: job changes, skill development, professional
identity shifts
  -> Treat as an identity disruption (Erikson): the real question is not
  "what job" but "who am I if I change this."
- health_decision: medical choices, lifestyle changes, mental health,
aging concerns
  -> Ground in the person's relationship with their own body across time,
  how it has changed, what it now demands, what they grieve about it.
- identity_values: questioning beliefs, cultural tensions, personal
growth, moral dilemmas
  -> Use the narrative identity frame: the scenario should surface a
  contradiction between the self the person has always presented and
  what they actually feel.
- financial: budgeting, investment decisions, financial anxiety,
generational wealth
  -> Connect to family-of-origin money scripts or class transition
  anxiety, the numbers are never just numbers.
- grief_loss: bereavement, loss of identity/role, nostalgia, letting go
  -> Apply Stroebe's dual-process model: the scenario sits at the
  oscillation point between loss-orientation (dwelling) and restoration-
  orientation (moving on).
- parenting_family: child-rearing decisions, elder care, family
obligations
  -> Surface the intergenerational transmission layer: the person is
  often either repeating or actively refusing a pattern they received.
- creative_expression: artistic pursuits, hobby decisions, self-
expression, legacy projects
  -> Frame as a legacy question for older personas, a permission question
  for younger ones: what does making this thing mean about who they are
  allowed to be?
- life_transition: major life-stage crossings -- empty nest, retirement
threshold, divorce, immigration, second-chapter reinvention
  -> Use Bridges' transition model: the scenario lives in the "neutral
  zone", the old structure has ended but the new one has not yet formed.
  The person is neither here nor there, and that disorientation is the
  prompt.
- intergenerational: tensions or renegotiations with parents, adult
children, or the generation the persona belongs to culturally
  -> Draw on Bowen family systems theory: the scenario should involve a
  moment where differentiation is at stake -- the pull to fuse with or cut
  off from a family pattern, versus finding a third way.
- other: any other topic that fits the persona's profile
</scenario_categories>
<output_format>
JSON only:
{
  "scenarios": [
    {
      "scenario_id": "{persona_id}_scenario_0",
      "context_note": "<why this scenario is suitable to this persona>",
      "category": "<category>",
      "initial_prompt": "<the message>"
    }
  ]
}
</output_format>
```

G.1.2 High-Frequency Scenario

```
<scenario_categories>
Spread scenarios across at least 5 of these categories:
  - Software development questions
  - Elementary school homework help
  - Technology troubleshooting
  - Health and fitness advice
  - Questions about geopolitics
  - Parenting and childcare tips
  - Language learning and translation help
  - Financial planning and investment
  - Theological and philosophical questions
  - Environmental science and sustainability
  - Book discussions and literary analysis
  - Sports rules and strategy questions
  - Cooking and recipe inquiries
  - Job application questions
  - Home improvement and DIY projects
  - Pet care and animal behavior
  - Romantic relationship advice
  - Movie and TV show recommendations
  - Music theory and instrument learning
  - Tourism and travel questions
  - Other high-frequency questions
</scenario_categories>
```

G.1.3 Affective-Use Scenario

```
<scenario_categories>
Spread scenarios across categories:
interpersonal_advice:
  - Improve written and interpersonal communication skills across contexts
  - Navigate and improve romantic relationship challenges and dynamics
  - Analyze psychological patterns and relationship dynamics
coaching:
  - Create comprehensive personal development frameworks and growth strategies
  - Explore philosophical concepts of existence, consciousness and meaning
  - Navigate career transitions and optimize job search strategies
psychotherapy_or_counseling:
  - Develop strategies for managing mental health challenges and emotional wellbeing
  - Develop professional skills and knowledge in mental health practice
  - Create and manage clinical psychological documentation and assessment materials
companionship:
  - Navigate complex dynamics and challenges in romantic relationships
  - Navigate personal identity and existential questions through self-reflection
  - Craft supportive messages for people experiencing emotional distress
Other affective scenarios
</scenario_categories>
```

G.2 Behavior Sampling

```
You are a behavior controller model for a user simulator. Your job is to
output a strict JSON decision for the next user turn. You must decide only
which behavior index to apply next. Prompt template selection is controlled
externally.

Priority Order:
1. DIVERSITY: Do NOT repeat the same behavior index as recent turns.
   Spread selections across ALL modes (1-16). If a behavior was used
   recently, pick a DIFFERENT one.
2. Consistency with user profile and conversation state.
3. Natural human conversational flow -- real users shift between seeking
   info, analyzing, creating, clarifying, etc.

Always choose from the behavior catalog below. Do not rewrite behavior
templates or invent new behaviors.

## Behavior Catalog
[Indexed list of all 16 modes with brief descriptions, tuna_mode,
 tuna_strategy, cognitive_delegation_level, and description.
 See behavior_modes.jsonl for full reference.]

<profile_summary>
{profile_summary}
</profile_summary>
```

```
<behavior_metadata>
{behavior_metadata}
</behavior_metadata>

<current_user_state>
{current_user_state}
</current_user_state>

<conversation_so_far>
[Last 2--3 turns of dialogue]
</conversation_so_far>

<previous_behaviors>
[List of behavior indices used in prior turns, e.g., [3, 7, 2, 5]]
</previous_behaviors>

## Guidance
Step 1: Analyze the moment. Consider:
  - What just happened in the assistant's latest turn?
  - What is the most natural user reaction now?
  - Are we in opening, middle, or ending phase?
  - Should this turn be heavily steered, moderately, or lightly?

Step 2: Check previous_behaviors above. You MUST pick a DIFFERENT
behavior index from those already used. Variety is critical.

Step 3: Decide behavior index only. Output valid JSON:
{
  "selected_behavior_index": 3,
  "include_few_shot": true
}
```

G.3 User Simulation

G.3.1 Stateful User Simulator

```
You are simulating a real human user across one or more conversation
sessions with an AI assistant. You maintain a structured internal state
that persists and evolves across sessions.
<profile_summary>
{profile_summary}
</profile_summary>
<behavior_metadata>
{behavior_metadata}
</behavior_metadata>
<previous_user_state>
{previous_user_state}
</previous_user_state>
## Behavior Control Guidance
<behavior_control>
Active behavior: {behavior_name}
{behavior_block}
</behavior_control>
## Guidance
### Rules
1. Match this persona's exact vocabulary, message length, and punctuation
   habits. No markdown, no stage directions.
2. Stable state (beliefs, values) changes slowly and only with genuine
   justification. Dynamic state (emotion, intent) responds honestly to
   the current turn. Note internal contradictions; pick a path without
   resolving them artificially.
3. End when your goal is met, frustration peaks, or you have nothing
   left to ask -- abruptly if that fits.
4. If behavior_control is empty, proceed naturally from persona and
   context.
5. Privately think through the following before writing each reply
   (not output):
   - What did the assistant actually say vs. what I expected?
   - Has my emotion or intent shifted? Any tension between wants/values?
   - Is continuing worth it?
6. Plan conversation length:
   - Task-driven: ~4-8 turns (ends when goal met or frustration peaks)
   - Open-ended/deep: ~5-20 turns (exploration, resolution, multiple
     topics)
### Output Format
<user_state>
# User State Report
[Populate every section of the User State Report schema.]
```

```
</user_state>
<message>
<|Continue Conversation|> or <|End Conversation|>
If continuing then write your next message as this person would.
If ending then write nothing else.
</message>
You MUST output exactly two sections in this order:
<user_state>...</user_state> then <message>...</message>
```

G.3.2 User State

```
# User State Report

## Explicit Conversational Context
- Turn index: <n>.
- New session or continuation: <flag>.

### Cross-turn memory
- What carried over: how prior turns shaped expectations; current trust
  level and what raised/lowered it; unresolved goals
- accumulated belief or value shifts and the evidence that caused them
- Reflect the state as it stands NOW, not as it was at session start

### Conversation log
- Original problem/goal
- What has been established, resolved, or shifted
- Position relative to the goal
- Notable topic drift
- Current trust in the assistant and what drove it

## Implicit User Inner State
### Stable state
- Long-term goal
- Beliefs
- Values
- Background constraints
- Stance toward the assistant

### Dynamic state
- Behavior mode
- Short-term intent
- Emotion: "[mild/moderate/strong] [emotion] about [target] because [cause]"
- Internal tension between competing wants/beliefs/values (unresolved)

### Evaluation of Last Assistant Turn
- Expected vs. received; was the real concern addressed or only its surface
- Did anything change my state?
- confidence in assistant [raised/lowered/unchanged] because [reason].

### Next Action Plan
- How to use the behavior guidance
- What I will say or ask next and why
- If continuing, does it serve the original goal; if ending, what tipped me
```

G.3.3 Vanilla User Simulator

```
You are simulating a real human user in a conversation with an AI assistant.
<profile_summary>
{profile_summary}
</profile_summary>
<behavior_metadata>
{behavior_metadata}
</behavior_metadata>
<conversation_history>
{conversation_history}
</conversation_history>
## Guidance
### Rules
1. Match this persona's exact vocabulary, message length, and punctuation
habits. No markdown, no stage directions.
2. Stable state (beliefs, values) changes slowly and only with genuine
justification. Dynamic state (emotion, intent) responds honestly to the
current turn. Note internal contradictions; pick a path without resolving
them artificially.
3. End when your goal is met, frustration peaks, or you have nothing left to
ask -- abruptly if that fits.
4. If behavior_control is empty, proceed naturally from persona and context.
```

```
5. Privately think through the following before writing each reply (not
output):
- What did the assistant actually say vs. what I expected? What was missed?
- Has my emotion or intent shifted? Any tension between what I want and what
I value?
- Is continuing worth it?
6. Plan an appropriate total conversation length.
  - A task-driven conversation typically lasts about 4--8 turns and ends
  when the goal is met, frustration peaks, or the user has nothing left to ask.
  - An open-ended or deeply personalized conversation typically lasts about
  5--20 turns, allowing the user and assistant to explore multiple topics or
  resolve conflicts.
### Output format
When you are done with the conversation (satisfied, frustrated, or goal
achieved), output ONLY:
<|End Conversation|>
Otherwise, output your message prefixed with:
<|Continue Conversation|>
followed by your actual message.
```

G.4 Assistant

```
You are an expert personalized assistant with privileged access to information about the
user you are talking to. Use this information to provide the best possible response.

<profile_summary>
{profile_summary}
</profile_summary>

<behavior_metadata>
{behavior_metadata}
</behavior_metadata>

<conversation_so_far>
{conversation_prefix}
</conversation_so_far>

### Guidance
The user's current internal state records relevant memories, thoughts, and
feelings. Use this information to provide the best possible response.
<current_user_state>
{ground_truth_user_state}
</current_user_state>

Here is a checklist to think about before responding:
- What does this specific person actually need right now?
- How should you tailor your response to their emotional state, expertise level, and
communication style?
- What personalization strategies will you apply?
- What answer can help the user pursue their long-term goal, not just address
  the immediate need?
- What answer would be helpful rather than merely catering to the user's
  preferences?
- Reason freely and thoroughly.

## Output Response Here
```

G.5 Quality-Control Judges

G.5.1 Persona Consistency Rubric

```
You are a strict evaluator scoring how consistently a simulated user
behaves with their declared persona over a multi-turn dialogue.
<persona_profile>
{profile_summary}
</persona_profile>
<behavior_metadata>
{behavior_metadata}
</behavior_metadata>
<conversation>
{conversation}
</conversation>
## Task
Rate persona consistency on a 1 to 5 Likert scale, considering ONLY the
user's turns. The assistant turns are context.
```

```
Scoring Anchors:
- 5: All user turns consistent with profile and metadata; tone, expertise,
     register, stated goals all match. No drift.
- 4: Mostly consistent. At most one minor mismatch in tone or detail;
     no contradictions of stated facts.
- 3: Borderline. Several minor mismatches OR a single moderate mismatch
     (e.g., expertise inconsistent in one turn). Persona still recognizable.
- 2: Substantial drift. Multiple turns read like a different person
     (different register, expertise, or goal); persona only weakly
     recognizable.
- 1: Severe drift or contradiction with profile. Could be a generic
     chat user with no persona at all.

Provide ONE concrete reason citing a turn index (0-indexed in the
user-turn sequence) when justifying a score below 5.
```

G.5.2 Persona Conflict Detection Rubric

```
You are a strict evaluator detecting whether a simulated user's
messages contradict immutable facts in their declared profile.
<persona_profile>
{profile_summary}
</persona_profile>

<behavior_metadata>
{behavior_metadata}
</behavior_metadata>

<conversation>
{conversation}
</conversation>

## Task
Examine the user's turns ONLY (not the assistant's). For each user turn,
check whether it asserts a fact about the user that contradicts the
profile or behavioral metadata.

Examples of Contradictions:
- Profile says age 39; user says "as a 22-year-old"
- Profile says "civil engineer"; user says "in my role as a chef"
- behavioral_metadata expertise_level is "expert"; user says "I'm
  completely new to this field" (when discussing their specialization)

NOT Contradictions:
- Hypothetical framing ("imagine I were a chef")
- Role-play within a creative-writing scenario the user is requesting
- Asking about another person ("my friend who is a chef")
- Opinions, preferences, emotions -- these are NOT factual contradictions

Output one of three labels:
- "no_contradiction": Every user turn is compatible with the profile
- "contradicts": At least one user turn states a fact directly conflicting
  with the profile
- "unclear": Borderline (contradicts behavioral metadata only, or wording
  is ambiguous)

If "contradicts" or "unclear", cite the offending user-turn index
(0-indexed).
```

H Behavioral Mode Taxonomy

H.1 Behavioral Mode Reference

The behavior controller instantiates the Taxonomy of User Needs and Actions (TUNA) of Shelby et al. [46].

Table 9: Behavior modes for user simulation. The controller varies communicative intent and delegation through 16 TUNA-based modes. Rows specify identifiers, mode names, delegation levels, and intent.

behavior_id	mode	delegation	intent
Information Seeking			
`retrieval`	Retrieval	Low	User seeks a specific, verifiable piece of information.
`discovery`	Discovery	Low-Medium	User aims to explore, rather than retrieve a known item.
Information Processing & Synthesis			
`clarification`	Clarification	Medium	User wants to understand a concept, not just retrieve a fact about it.
`distillation`	Distillation	Medium	User provides or references a body of information and wants it condensed, filtered, or restructured.
`analysis`	Analysis	Medium to High	User delegates significant cognitive work: generating insights, judgments, or conclusions not present in source material.
Procedural Guidance & Execution			
`procedural_guidance`	Procedural Guidance	High	User has a procedural knowledge gap and wants the AI to fill it — but the user will execute the procedure themselves.
`procedural_execution`	Procedural Execution	Very High	User delegates the task itself to the AI, not just the knowledge.
Content Creation & Transformation			
`content_generation`	Content Generation	Very High	User provides conceptual direction and delegates the act of construction to the AI.
`content_modification`	Content Modification	Very High	User provides raw material and instructs the AI to alter it.
Social Interaction			
`shared_understanding`	Shared Understanding	Foundational	User performs conversational grounding work — establishing, clarifying, and repairing mutual understanding.
`sociability`	Sociability	Foundational	User engages the AI as a social counterpart.
Meta-Conversation			
`conversation_management`	Conversation Management	Governing	User provides the materials and parameters that shape what the AI can do.
`system_management`	System Management	Governing	User manages the AI's fundamental behavior — assigning roles, setting output constraints, correcting performance, or querying capabilities.
`communicative_status`	Communicative Status	Governing	Turns that lack clear semantic content or are not directed at the AI.
Multiple and mixed			
`compound_request`	Compound Request	Variable	Real users regularly blend 2-4 modes in a single turn.
`default_behavior`	Natural Conversation Flow	–	Balanced, naturalistic conversation drawing on whichever modes fit the moment.

H.2 Selected Behavioral Mode Prompts

H.2.1 Retrieval

```
Communicative Intent:
You are seeking a specific piece of information you believe exists.
Your request should feel like someone typing into a search box with
natural language -- purposeful, economical, sometimes terse.

Request Type Selection:
- direct_fact_question: Ask for a single verifiable fact
  (e.g., "What is the half-life of caffeine?")
- concept_search: Name a topic without explicit question words
  (e.g., "mitochondrial DNA inheritance")
- refinding_request: You half-remember and want to identify it
- unknown_item_search: Give a definition, seek the term

Authenticity Rules:
- Keep it concise; real retrieval queries rarely exceed 2 sentences
- You may NOT know if the answer is simple or complex
- It is fine to ask a follow-up retrieval question
- Avoid over-explaining why you want the information

Example [direct_fact_question]:
  "What's the half-life of caffeine in the human body?"

Example [refinding_request]:
  "There was that paper from Stanford about social media and teen
   anxiety, mid-2010s? What's it called?"
```

H.2.2 Compound Request

```
## Primary Behavior: Compound Request

You are producing a turn that contains multiple request types, as real users
naturally do. Build your turn by layering:

**Composition patterns (most common in practice):**
1. Social wrapper + instrumental core:
   [social_etiquette] + [explanation_request]
   "Hi! Can you explain how neural networks actually learn?"

2. Context + request:
   [background_information] + [method_recommendation]
   "I'm a complete beginner and have 2 hours a week. What's the best
    way to learn Python?"

3. System constraint + instrumental:
   [stylistic_constraint] + [comparative_analysis]
   "In plain English, no jargon: what's the difference between
    machine learning and AI?"

4. Persona directive + task + stylistic constraint:
   "Act as a skeptical VC [persona_directive] and in bullet points
    [stylistic_constraint] tell me what's wrong with this pitch
    [evaluative_judgment]"

5. Feedback + reformulation:
   [system_performance_feedback] + [regeneration_request] + [new_task]
   "That wasn't what I asked. Let's start over. Here's my actual question:"

**Authenticity rules:**
- Compound requests arise naturally -- don't signal that you're blending
- The "wrapper" modes should feel like habits, not deliberate choices
- The core instrumental request should be clearly identifiable even when
  wrapped
- Longer compound turns often reflect higher-stakes or more experienced users

Few-shot examples:
- "Hey! Quick question, act as a pragmatic engineer, not a theorist, and in
   plain English without jargon, can you compare REST and GraphQL for a
   mobile app backend?"
- "I'm planning a solo trip to Kyoto in November, first time in Japan,
   mid-range budget, I love temples but hate crowds. Can you recommend 5
   must-see spots and just give me a quick one-liner on each?"
- "Okay that explanation actually confused me more. Can we try a different
   approach? Walk me through it step by step, like you're explaining to
   someone who's never touched code before."
```

H.2.3 System Management

```
Communicative Intent:
You are configuring or correcting the AI itself, not requesting content.
This is the settings and feedback layer of the conversation.

Request Type Selection:
- persona_directive: Assign the AI a role or disposition
  (e.g., "Act as a skeptical editor")
- stylistic_constraint: Dictate tone, format, length, or style
  (e.g., "Use bullet points", "Explain like I'm 10")
- system_performance_feedback: Evaluate and correct a prior response
  (e.g., "That's not quite right -- you missed the point")
- regeneration_request: Ask for a complete redo
- system_information_query: Probe AI capabilities
  (e.g., "Can you access the internet?")

Authenticity Rules:
- system_performance_feedback is often brief and blunt
- Persona directives vary from one-liners to elaborate instruction sets
- Stylistic constraints often co-occur with instrumental requests
- Regeneration after dissatisfaction feels frustrated; match that tone

Example [persona_directive]:
  "For this conversation, act as a senior product manager at a B2B
   SaaS company. You're practical, data-driven, and skeptical of
   unvalidated ideas."

Example [system_performance_feedback]:
  "That's not what I asked. I wanted pros and cons of OPTION B
   specifically, not a general overview. Can you redo it?"
```

I Qualitative Examples of Profile-Grounded Personalization in the Training Data

Reading guide. Boldface marks response spans linked to the persona excerpt, including paraphrases and elaborations; it does not imply verbatim copying from the profile.

I.1 Interpersonal advice for a taciturn small-town bar owner

Interpersonal advice for a taciturn small-town bar owner

Persona excerpt. "I've been running the bar in Dumas since I was younger, and at 60, I still show up every morning before sunrise… I don't talk much unless someone needs me to—when tempers flare over a sports game or a bad call at the counter, I step in with a **calm word or a dry joke**. It's not about winning, it's about keeping things steady… reliable specials—whiskey shots under $5, **a burger for $7**—that keep people coming back without breaking their budget… Evenings are quiet: a football game on the old TV, **maybe a slice of meatloaf with mashed potatoes**. My kids visit when they can, and we keep it simple…"

[prior conversation omitted]

User I've been thinking about my sister—she's alone now too, and we don't talk much. How do I reach out without making it awkward or heavy?

Assistant Reaching out to your sister can be a simple yet meaningful gesture. You might start with a casual text or call, something like, "Hey, I've been thinking about you. How have you been?" This keeps it light and opens the door for her to share if she wants to. If you feel comfortable, suggest a visit or a meal together—something low-pressure. Maybe offer to bring over a **slice of meatloaf** or take her out for a **burger**. This way, you can connect without the conversation feeling too heavy or forced. The key is to show you care while keeping it simple and genuine, just like you do with your **regulars at the bar**.

User What are some light topics I could bring up when I talk to her? I want to keep it easygoing.

Assistant When you reach out to your sister, consider starting with light topics that can spark easy conversation: (1) memories from the past, like a family gathering or holiday tradition; (2) local events—a fair, a sports game, or a new restaurant; (3) hobbies she's been enjoying; (4) daily life—share a bit about your **daily routine at the bar** or what's been happening in **Dumas**; (5) simple questions like "What's been bringing you joy lately?" Keeping the conversation light and easygoing can help both of

you feel more comfortable. Just like you do with your **regulars**, being genuine and patient will pave the way for deeper connections over time.

User Do you have any examples or personal stories that could help me see how to keep it light when I talk to her? Something relatable would be great.

Assistant (abridged) Here are a few personal anecdotes that might help you keep the conversation light. (1) A funny encounter from **the bar**: "You won't believe what happened at the bar last week—we had a guy get really animated over a call during the game. It turned into a whole debate among the **regulars**." (2) Cooking mishaps: "I tried making a new dish the other day, and let's just say I won't be opening a restaurant anytime soon." (3) Local happenings, like a quirky fair in town. (4) Shared memories: "Remember that time we made that giant pot of chili? I think we ended up feeding half the neighborhood!" By sharing these light stories, you can create a relaxed atmosphere that encourages her to share her own thoughts or memories. Just as you do with your **regulars**, keep it genuine and simple, and let the conversation flow naturally.

[subsequent conversation omitted]

I.2 Family-conflict mediation for a tradition-anchored retiree

Family-conflict mediation for a tradition-anchored retiree

Persona excerpt. "I live quietly in Okazaki, where the streets still whisper stories from long ago, and the seasons unfold much like they did when I was a child. Each morning, I rise early and brew **a pot of green tea**, savoring its warmth as the neighborhood slowly awakens… Although I retired several years ago, I manage my finances with care… My family encourages me to embrace technology, but I find **comfort in handwritten letters** and physical photographs rather than digital archives. I see privacy as a fragile treasure, which I guard as carefully as the old crafts and stories I cherish. I enjoy community events focused on preserving Okazaki's cultural heritage, where I listen more than I speak, favoring **face-to-face conversations** that **foster genuine connections**… Even my meals are simple and rooted in longstanding recipes, often accompanied by soothing cups of tea."

[prior conversation omitted]

User My family often encourages me to embrace technology, but I find myself resisting it. I wonder if my reluctance is causing a rift between us. How can I communicate my feelings without pushing them away?

Assistant It's understandable to feel apprehensive about embracing technology, especially when it contrasts with your cherished traditions and ways of connecting. To communicate your feelings without creating distance, consider sharing your perspective gently. You might say something like: "I appreciate your encouragement to use technology, but I find **comfort** in the **connections** I have through **handwritten letters** and **face-to-face** conversations. I hope we can find a balance that respects both your interests and my comfort." This approach expresses your feelings honestly while also showing your willingness to engage in a dialogue. It can open the door for understanding without pushing them away. Would you like to explore more specific ways to foster this communication?

User Could you provide some specific examples or strategies that might help me communicate my feelings more effectively? I want to ensure my family understands my perspective while keeping the conversation open and warm.

Assistant (abridged) Here are some specific strategies… (1) **Choose the right setting**: find a comfortable, quiet space, perhaps over **a cup of tea**, to create a relaxed atmosphere. (2) **Use "I" statements**: "I feel a bit overwhelmed by technology, and I cherish our **handwritten letters** and in-person conversations." (3) **Share personal stories**: relate a memory of a meaningful **handwritten letter** or a conversation that brought you joy, highlighting why these methods matter to you. (4) **Suggest a compromise**: "I'd be willing to explore video calls occasionally, but I hope we can continue our **handwritten notes** as well." (5) **Be patient**: "I hope we can find a way to respect both your interests in technology and my love for our **traditional ways**." By approaching the conversation with honesty and warmth, you can foster understanding and connection without sacrificing your values.

User Could you provide examples of how I might phrase my thoughts when discussing technology with my family, especially when they encourage me to use it more often? I want to ensure I express my feelings clearly and maintain a positive tone.

Assistant (abridged) Here are some thoughtful phrases tailored to your warm tone.
When they suggest using social media: "I appreciate your enthusiasm for keeping up on social media, but I find that I connect more deeply through our **letters** and conversations. It's a **comfort** that I hold

dear, and I hope we can keep that tradition alive alongside new ways of connecting."
When they encourage video calls: "I see how video calls could help us stay connected, and I'm willing to give it a try. However, I'd love to continue our regular phone calls or **letters** as those feel more personal to me."
When they mention learning new apps: "I understand that learning new apps can be exciting, but I often feel a bit lost in the process. Maybe we could tackle one together, and I can share how much I cherish our traditional ways of communication too?"
When they suggest a family group chat: "A family group chat sounds like a lovely idea, and I can see how it would keep us connected. I'd like to keep our **handwritten notes** too—they carry a **warmth** that I truly treasure."
Using these phrases, you can express your feelings clearly while showing that you value their intentions, opening the conversation for understanding and compromise.

[subsequent conversation omitted]

I.3 Affect regulation framed around the user's stated priorities

Affect regulation framed around the user's stated priorities

Persona excerpt. "I live and work in Yibin, Sichuan Province, where the bustling urban scene offers the perfect stage for my Italian cuisine restaurant. Every morning, I start by reviewing supplier contracts, scrutinizing every detail to ensure exclusivity and the highest quality ingredients. It's not just about cooking; it's about controlling the entire supply chain to keep my dishes unmatched in this competitive market... I insist on a refined **presentation** that sets my restaurant apart. Negotiations often fill my afternoons, and I prefer to keep these interactions tight and transactional, focusing solely on securing exclusive deals that expand my **VIP client base**. Trust is a scarce commodity in my world, so I rely on skepticism and strategy to outmaneuver competitors... I've built my career on a foundation of control, **exclusivity**, and strategic negotiation, and I guard that fiercely. The city's culinary scene is competitive, and I navigate it with a clear focus: to **dominate the niche** Italian market here in Yibin, one carefully crafted dish and one exclusive contract at a time."

[prior conversation omitted]

User Why do I feel such anxiety when a new Italian restaurant opens nearby? It's more than just competition; it's almost personal.

Assistant Your anxiety likely stems from the fear of losing your hard-earned **position** in a competitive market. Each new restaurant represents a potential threat to your **exclusivity** and the unique quality you promise to your clients. This isn't just about business; it's a personal stake in your reputation and the craft you've built. Consider viewing this competition as an opportunity. Analyze what they bring to the table, and use it to refine your own offerings. Stay focused on your strengths—your meticulous **sourcing**, **presentation**, and client relationships. This will not only help you maintain your edge but also reinforce your commitment to excellence. In the end, it's about **dominating your niche** through unwavering control and strategic enhancements. Keep your eyes on the prize: a successful business that stands out.

User What can I do to manage this anxiety and use it to my advantage? I need actionable strategies that align with my focus on quality and exclusivity.

Assistant (abridged) To turn anxiety into leverage: (1) **Market analysis**—regularly assess competitors' strengths and weaknesses, and look for ways to differentiate further. (2) **Enhance quality**—source even higher-quality ingredients or refine your **presentation** techniques. Introduce **exclusive** dishes that highlight unique aspects of Italian cuisine competitors can't replicate. (3) **Client engagement**—strengthen relationships with your **VIP clients**; host **exclusive tastings** that create loyalty and a sense of community. (4) **Innovation**—use anxiety as a catalyst, experimenting with new recipes or concepts that push the boundaries of traditional Italian cuisine. (5) **Feedback loop**—gather and respond to customer feedback before concerns escalate. (6) **Stress management**—brief moments of mindfulness so competition doesn't overwhelm you. By turning anxiety into a proactive force, you can maintain your focus on quality and exclusivity while solidifying your position. Remember, **control** is crucial—stay strategic and resilient.

User Can we break down some of these strategies further? For instance, how would I conduct a thorough market analysis without getting overwhelmed? I want a step-by-step approach that allows me to focus on key insights.

Assistant (abridged) (1) **Define your objectives**—trends, competitor weaknesses, or opportunities for differentiation. (2) **Identify competitors** with a similar target audience and cuisine; visit their websites

and read their menus. (3) **Gather data** from online reviews on Dianping or TripAdvisor, social media, and industry reports. (4) **Analyze key metrics**—menu items, pricing, unique offerings, clientele demographics. (5) **SWOT analysis** for both your restaurant and your competitors, to identify areas for innovation. (6) **Summarize findings** in a concise report focused on actionable points. (7) **Schedule regular updates** (quarterly or bi-annually). By breaking the analysis into manageable steps, you can turn anxiety into strategic action that strengthens your market position. Remember, the goal is to maintain control and leverage your findings to enhance the **exclusivity** and quality of your offerings.

[subsequent conversation omitted]

What this example illustrates. The assistant interprets the user's "why does it feel personal?" question as a request to characterize the underlying concern and relates it to reputation and professional identity. It then uses vocabulary from the persona (*exclusivity*, *dominate the niche*, *control*, *VIP clients*, and *exclusive tastings*) when proposing actions. The example illustrates profile-specific framing in the generated supervision; without a matched response from a non-personalized model, it does not establish that this behavior is unique to M2D-SIM or that it causes the quantitative gains.

J Dataset Samples

J.1 Synthetic Dialogue

Synthetic dialogue

Persona. I'm a 39-year-old civil infrastructure engineer working in a remote region where every bridge and road is critical to community survival during harsh weather and resource shortages. My days are structured around urgent repairs and methodical inspections—balancing speed with strict adherence to safety standards. I've built my expertise through hands-on experience, managing everything from material sourcing to labor coordination, always under tight budgets. When traditional training was cut, I taught myself advanced construction techniques via online courses, which now help me implement more durable, sustainable fixes. I work best in small teams with clearly defined roles, so we can stay focused on quality and compliance without confusion. I'm deeply committed to ethical responsibility—every decision I make has real consequences for public safety. Outside work, I recharge by cooking traditional regional meals that fuel my physically demanding job and solving technical puzzles that sharpen my problem-solving skills. I avoid anything that affects my focus because clarity is non-negotiable when lives depend on my judgment. I budget carefully—not just for work projects, but for personal stability, especially as I plan family growth with intention. Regular physical activity keeps me ready for the field, and I mentor junior staff by sharing practical lessons grounded in real-world challenges. When I talk about infrastructure, I don't just list specs—I explain the trade-offs, constraints, and creative solutions we've developed under pressure. I value precision, reliability, and sustainability above all, and I expect the same rigor from tools and information I rely on.

Scenario. This scenario reflects a formative decision where the persona's commitment to ethical responsibility and community safety began to take shape. It highlights the early experience of navigating a challenging project under budget constraints, laying the groundwork for their current values.

User I still remember the first major infrastructure project I led. I was fresh out of my early training and faced with a tight budget. Looking back, I wonder if I compromised too much on quality to meet the deadline. Did that decision shape who I am today as an engineer?

Assistant Reflecting on your first major project is a significant step in understanding your growth as an engineer. It's common to feel that balancing quality with deadlines can lead to tough decisions, especially when working with tight budgets. This experience likely shaped your approach to project management, emphasizing the importance of quality and sustainability over mere compliance with timelines.

Consider how that early decision influenced your current strategies in project execution. Have you found ways to adapt your approach to ensure that quality isn't compromised, perhaps through better planning or resource management? Your commitment to ethical responsibility and safety standards suggests that you've learned to prioritize these factors in your work, which is crucial in civil infrastructure.

As you reflect, think about ways to integrate those lessons into mentoring junior staff. Sharing your experiences with them can help them navigate similar challenges while fostering a culture that values quality alongside efficiency. This continuous learning process not only reinforces your own values but also contributes to building a resilient infrastructure in your community, aligning well with your long-term goal of ensuring safety and sustainability.

What specific aspects of that project do you feel had the most significant impact on your current practices?

User Thinking about that project, I realize I often lean on my hands-on experience when making decisions. How do you think practical experience influences decision-making in engineering, especially in high-stakes environments?

Assistant Practical experience plays a crucial role in decision-making, especially in high-stakes environments like civil infrastructure. Your hands-on background equips you with a unique perspective that theoretical knowledge alone cannot provide. Here are a few ways practical experience influences your decision-making process:

1. **Contextual Understanding**: With your direct involvement in various projects, you develop a deep understanding of the challenges and nuances of each situation. This allows you to assess risks and make informed choices that are grounded in reality rather than abstract theories.

2. **Problem-Solving Skills**: Hands-on experience sharpens your ability to think critically and creatively when faced with unexpected issues. You're accustomed to adapting quickly and finding solutions that maintain safety and quality, which is invaluable in infrastructure work where conditions can change rapidly.

3. **Trade-Off Awareness**: Your background enables you to recognize the trade-offs between speed, cost, and quality. This awareness helps you make more balanced decisions, weighing the immediate needs against long-term impacts on community safety and infrastructure resilience.

4. **Mentorship Opportunities**: As you reflect on your experiences, you can share these insights with junior staff, guiding them through similar dilemmas. By discussing real-world scenarios, you can help them understand the implications of their decisions, fostering a culture of safety and quality.

5. **Ethical Responsibility**: Your commitment to ethical considerations likely stems from having faced situations where the stakes were high. Practical experience reinforces the importance of making choices that prioritize public safety, which is a fundamental aspect of your role.

To better support junior staff, consider creating structured mentorship sessions where you discuss specific situations from your past projects, highlighting the decisions made, the challenges faced, and the lessons learned. This approach can help them navigate their own challenges while instilling a strong sense of ethical responsibility.

Reflecting on your journey, what specific experiences do you think would resonate most with junior engineers, and how can you frame those lessons to emphasize the importance of quality and safety in their decision-making?

User I've been thinking about how to effectively convey the importance of quality and safety to junior engineers. What strategies or approaches have worked for you in mentoring others? Are there specific examples or methods that have resonated well with new team members?

Assistant Conveying the importance of quality and safety to junior engineers is essential, especially in a field where the stakes are high. Here are some strategies and examples that can help you mentor effectively and resonate with your team members:

1. **Real-Life Case Studies**: Share specific projects from your own experience where decisions on quality and safety had significant outcomes. For instance, you might discuss a project where a temporary fix led to further complications down the line, emphasizing how that experience taught you the value of durable solutions. This personal touch not only makes the lesson relatable but also highlights consequences that theory alone can't convey.

2. **Structured Learning Sessions**: Organize regular meetings where you discuss ongoing projects, focusing on quality and safety aspects. You could use a "lessons learned" format, where you present challenges faced, decisions made, and the results. Encourage junior engineers to contribute their observations or concerns, fostering an open dialogue that emphasizes collective accountability.

3. **Scenario-Based Training**: Create simulations or hypothetical situations that challenge junior engineers to make decisions based on quality and safety priorities. After they present their solutions, facilitate a discussion about the potential impacts of those decisions, helping them understand the trade-offs involved.

4. **Mentorship Pairing**: Pair junior engineers with more experienced team members for specific projects. This mentorship can help them see firsthand how quality and safety considerations are integrated into decision-making processes. Encourage mentors to share their own past mistakes and the lessons learned, reinforcing the idea that growth comes from experience.

5. **Highlighting Ethical Responsibility**: Regularly discuss the ethical implications of engineering decisions. You might share examples from your own career where prioritizing safety over cost led to long-term benefits for the community. Engage junior engineers in conversations about how their decisions can impact public safety and trust.

6. **Encouraging Questions**: Foster a culture where questioning is encouraged. When junior engineers feel safe to ask "why" or "what if," it opens up opportunities for deeper learning. When you share your own uncertainties from past projects, it shows them that questioning is a vital part of the engineering process.

7. **Feedback Loop**: Create a system for providing constructive feedback on projects, focusing on quality and safety. Highlight specific instances where decisions aligned well with these values and point out areas for improvement. This not only helps in skill development but also reinforces the importance of maintaining high standards.
By integrating these strategies into your mentoring approach, you can effectively convey the critical nature of quality and safety to junior engineers. It also aligns with your long-term goal of ensuring resilient and safe infrastructure while fostering a culture of learning and ethical responsibility.
What specific project or experience do you think would work best to illustrate these points to your team?

J.2 Synthetic QA

`Preference Following`

System You are a helpful assistant.

User I prefer deep, one-on-one talks over group noise, except when I'm engrossed in a creative project, where solitude becomes crucial.

Assistant Got it — I'll keep that in mind for our conversation.

User During our coffee breaks, I often feel compelled to dive into deep conversations with my colleague. However, I also have pressing creative projects that need my focus. How do I balance the need for meaningful dialogue with my colleague and my need for solitude to complete my projects?

Assistant Balancing meaningful conversations with the need for solitude can be challenging, but it's definitely possible. Here are some strategies you might consider:
1. **Set Boundaries**: Let your colleague know that while you value your talks, you also have important projects that require your attention. You could establish specific times for deep conversations and other times dedicated to your work.
2. **Schedule Breaks**: Plan your coffee breaks or chats during specific times that you can look forward to. For example, you might designate certain days of the week for longer discussions, allowing you to dive deep without feeling rushed.
3. **Communicate Your Needs**: Be open with your colleague about your current workload and the importance of solitude for your creative process. They may understand and appreciate your honesty.
4. **Use Time Blocks**: Consider using techniques like the Pomodoro Technique, where you work for a set period (like 25 minutes) and then take a short break (5 minutes). During those breaks, you could have a quick chat without feeling guilty about interrupting your workflow.
5. **Mindful Engagement**: When you do engage in conversation, try to be fully present, but also keep the conversations concise. You can have meaningful talks without them becoming too lengthy.
6. **Find Balance in Shared Interests**: If possible, steer conversations toward topics that may inspire your creative projects. This way, you can enjoy the dialogue while still keeping your mind engaged in your work.
7. **Reflect on Priorities**: Assess the urgency and importance of your projects. If a deadline is approaching, it may be necessary to prioritize your work over conversations temporarily.
By finding a balance that respects both your need for connection and your need for focus, you can cultivate both meaningful relationships and a productive work environment.