THE OTHER SIDE OF THE CONVERSATION: LEARNING BEHAVIORAL USER MODELS FROM USER-SIDE TURNS

Anonymous authors
Paper under double-blind review

ABSTRACT

Conversational intelligence requires an assistant not only to respond to what is said, but to anticipate how a particular interlocutor is likely to respond. But can LLMs construct such a user model from dialogue alone? We study this question in a controlled setting where a synthetic user is defined by a combination of persona and style, encoded in a known system prompt P. We evaluate how well three strategies can approximate P by only observing conversations, without having access to P: 1) conditioning on demonstrations in context, 2) explicitly reconstructing a natural-language description of the user, and 3) learning a continuous soft prompt from user-generated turns. Our evaluation shows that these conversation-only methods recover a substantial amount of user-specific signal. However, fidelity to the user depends on the criterion we apply: distributional agreement, recovery of persona and style, or how reliably generated text is judged to come from a user. Our framework provides a quantitative way to evaluate user simulation from dialogue and its limits.

1 INTRODUCTION

During a conversation with someone, we convey information and advance our communicative goals, but we also model our interlocutor: their knowledge, goals, preferences, and how they will likely respond to us (Halliday & Matthiessen, 2013; Hovy & Yang, 2021). In humans, this model is governed by concepts like theory of mind (Wimmer & Perner, 1983) and Gricean maxims (Grice et al., 1975), and functions as the basis of successful conversations. For an LLM assistant to use similar conversational intelligence, they would need a predictive model of any particular user to simulate possible reactions before choosing its own response. However, existing work in this area mostly studies adjacent capabilities: *personalization*, by conditioning assistants on user histories (Salemi et al., 2024; Doddapaneni et al., 2024; Zhao et al., 2025), *user simulation*, by training models to mimic the human side of dialogue (Naous et al., 2026); and *theory-of-mind*, by testing whether LLMs reason about other agents' beliefs and knowledge (Sap et al., 2022; Kim et al., 2023; Chen et al., 2024). Inferring information about an interlocutor from dialogue is itself not new (Zhang et al., 2018), but a basic question remains open: *how faithfully can an LLM recover an interlocutor's response-generating distribution from the behavior they exhibit in conversation, and how could we quantify that recovery against a known reference?*

Answering this question is hardest where it matters most. There is no ground truth "real person" to exhaustively score an inferred user model against. We can use narrow questionnaires and held-out decisions to test particular predictions (Park et al., 2024), but those are individual probes: they cannot tell us how much of the user was available to recover in principle. Did the conversation just never reveal the relevant information, or did the learner fail to extract it? In both cases, the prediction would not succeed.

To make this distinction measurable, we rely on synthetic users. We construct 50 synthetic users by crossing ten personas from SCOPE-Persona (Venkit et al., 2026) with five three-feature writing styles from StyleDistance (Patel et al., 2025a). We use these persona-and-style prompts P as our user to condition a frozen LLM. This setup allows us to directly query the "user's" generative distribution

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

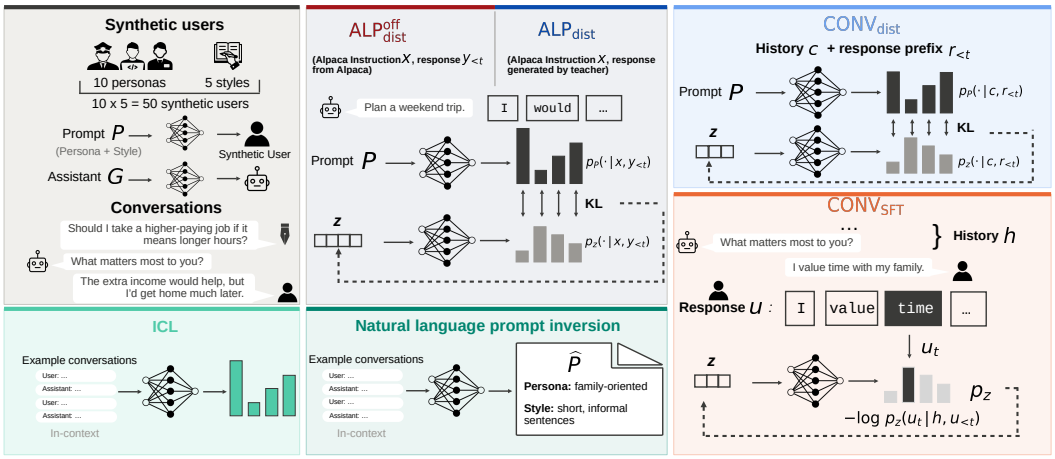


Figure 1: **Framework overview.** Persona–style prompts P define 50 **synthetic users**, whose conversations with a generic assistant provide the observational data. *Learning from teacher distributions* (privileged): ALP_{dist} matches the next-token distributions of the model under P on Alpaca instructions with teacher-generated responses ($\text{ALP}^{\text{off}}_{\text{dist}}$: same objective, responses from Alpaca itself), while $\text{CONV}_{\text{dist}}$ matches them on the observed conversations. *Learning from observations alone* (no access to P): **ICL** conditions on example conversations in context; natural language prompt inversion summarizes them into a description $\hat{P}$; CONV_{SFT} trains soft tokens z by next-token prediction on the user's responses u. All soft-prompt methods share capacity and template; only the supervision and training contexts differ.

and control for various confounding factors. We begin each conversation with a conversation starter (*dilemma* in the following) drawn from a curated set.

A separate instance of the same LLM, conditioned as a generic conversational partner, interacts with the synthetic user as the assistant. Our methods try to build a predictive model of the user's conversational behavior, but without having access to P. We compare three strategies to infer P: 1) in-context learning (ICL), which uses the observed conversations as demonstrations to condition on; 2) natural-language prompt inversion, which reconstructs an explicit prompt description of the user, $\hat{P}$, from model output text; and 3) soft prompt tuning, which learns a continuous prompt, z, trained only on the user-generated turns, by applying next-token prediction loss. This is a student-teacher setup. All three strategies rely on conversational evidence produced by P.

The known prompt P provides a common behavioral reference to evaluate all three strategies. We evaluate all user models on 16 held-out topics, measuring the symmetric KL divergence at each response token from P. For the soft-prompt strategy, we construct three additional measures that serve as teacher baselines. These capacity-matched references directly distill the next-token distributions P induces, instead of learning only from sampled tokens. The first baseline is an *on-policy* reference distilled from the same conversations used to learn z. The second distillation reference is still *on-policy* but trained on an external distribution of instruction contexts. The final baseline is instead *off-policy*: both instructions and responses used as context for the distillation are independent of P. Together, these baselines let us diagnose from where performance gaps arise: the available conversational evidence, the learning signal extracted from it, or how much information the prompt representation carries.

Our experiments show that the model can recover some salient patterns of a user's behavior *without any parameter updates*, through inference-only methods. With just a few examples (ICL) or with natural-language prompt inversion, it produces questionnaire answers consistent with the user and generates responses recognizable by an LLM judge as coming from the target user. This behavioral resemblance, however, can be just good impersonation, and does not imply faithful recovery of the user's conditional output distribution. For that, the model requires user-specific optimization on the observed user turns. We find that soft prompt tuning provides a lightweight and effective way to

perform exactly this optimization. This contrast shows that recognizable simulation of a user and predictive agreement are two distinct aspects of user modeling. These findings have implications for lightweight personalization approaches, but potentially also for privacy.

Code and configurations are available here.

Contributions

1. **A protocol that makes user-model recovery measurable.** With P known, we evaluate recovery against explicit references rather than estimated from prediction error. We are the first, to our knowledge, to separate out two limiting factors: what the conversations reveal about the user, and what each modeling strategy extracts from them.
2. **A quantitative account of conversational user modeling.** Optimizing on observed user turns approaches the conversation-distilled reference and outperforms the inference-only strategies. Recovery quality scales with conversational breadth and depth, and depends only weakly on representational capacity.
3. **Evidence that user-model fidelity is multi-faceted.** Distributional, geometric, and generative criteria rank the same models differently: inference-only strategies yield user-models that are recognizable without distributional fidelity, while training additional parameters allows better alignment even at the token level. The recovered models enable concrete uses, from user characterization to change detection.

2 Related Work

Personalization and user simulation. Existing research studies how a user's profile and conversational history can improve model outputs (Salemi et al., 2024; Doddapaneni et al., 2024), while persona-conditioned dialogue and user simulation emphasize generating user-like behavior (Zhang et al., 2018; Naous et al., 2026). Generative agents use descriptions and accumulated experience to condition behavior (Park et al., 2023). More directly, Park et al. (2024) construct individual-specific agents from interviews and evaluate agreement with participants' attitudes and behavior. Our aim is to quantify what conversation reveals about a user, rather than improve personalization. A controlled synthetic setting lets us measure behavioral recovery against a known response distribution.

Prompt inversion and user inference. Prompt inversion and extraction methods reconstruct hidden prompts from model outputs or next-token probabilities (Morris et al., 2024; Zhang et al., 2024; Gao et al., 2024). Related work demonstrates that LLMs can infer personal attributes from text, raising privacy concerns beyond memorization (Staab et al., 2024). We study functional recovery: learning a representation that reproduces user-specific response distributions without requiring reconstruction of the defining prompt or restricting evaluation to predefined personal attributes.

Continuous prompts and distillation. Prefix tuning learns continuous prefixes across model layers (Li & Liang, 2021), while prompt tuning optimizes input embeddings to adapt frozen models to downstream tasks (Lester et al., 2021). Wingate et al. (2022) compress known textual prompts into short continuous representations through distribution matching. We build on this formulation and characterize the recovered user models across how they are constructed (in-context conditioning, natural-language inference, or soft-prompt optimization) and the evidence they receive: sampled responses or teacher distributions, on conversational or external instruction contexts. This allows us to examine how inference strategy, supervision, and context distribution shape behavioral recovery.

An extended discussion is provided in Appendix A.

3 A Framework for Behavioral User Recovery

We define *behavioral user recovery* as the extent to which an LLM assistant can use a finite set of observed conversations to bring its conditional next-token distribution into agreement with that of a particular user. Figure 1 summarizes the framework, from synthetic-user construction and conversation generation to alternative strategies for learning a user model. We compare observational methods (training-free and optimization-based) with privileged distillation references to quantify how much representation, training contexts, and supervision affect recovery.

Synthetic users and conversational evidence. We construct 50 synthetic users by crossing ten personas from SCOPE-Persona (Venkit et al., 2026) with five writing styles, each defined by three features from StyleDistance (Patel et al., 2025a). Each persona–style prompt P conditions a frozen Qwen3.5-9B model (Team, 2026) through its system prompt. Conversations begin with a first-person dilemma from a curated set of size 216. A separate instance of the same model, conditioned on a generic system prompt, responds to this message; the synthetic user and assistant then alternate turns. Appendix B.1 details the construction and generation procedures. We extend our framework to a different model family, Olmo 3 (Olmo et al., 2025), in Appendix D.

Learning from conversational observations. We compare three strategies that construct a user model from observed conversations without using P or its next-token distributions as input or training targets. Two require no parameter updates. *In-context learning (ICL)* retains conversations verbatim in the system prompt. *Natural-language prompt inversion* asks the frozen LLM to reconstruct a system prompt describing the user's persona and style, $\hat{P}$, which replaces P during evaluation.

The third strategy, $\mathrm{CONV}_{\mathrm{SFT}}$, learns a separate soft prompt for each user. A soft prompt z consists of k trainable token embeddings placed after the fixed system prefix "You are an assistant." We use five tokens except for the corresponding ablation. For each observed user response u, the model receives the full preceding dialogue h and response prefix $u_{<t}$. Keeping the backbone frozen, we update only the soft-prompt embeddings to minimize

$$\mathcal{L}_{\mathrm{obs}}(z) = -\mathbb{E}_{(h,u)\sim\mathcal{C}_P}\left[\frac{1}{|u|}\sum_{t=1}^{|u|}\log p_z(u_t \mid h, u_{<t})\right], \tag{1}$$

where $\mathcal{C}_P$ contains the observed history–response pairs and p_z denotes the model's next-token distribution under the soft-prompt template. Both participants' preceding messages provide context, but the opening dilemma and the generic assistant's messages are not prediction targets.

Learning from teacher distributions. To calibrate observational recovery, we construct three privileged soft-prompt references by distilling the frozen model conditioned on P. They all share the backbone, system-prompt template, and representation capacity of $\mathrm{CONV}_{\mathrm{SFT}}$, but differ in their training contexts. Following Wingate et al. (2022), each minimizes

$$\mathcal{L}_{\mathrm{dist}}(z; P, \mathcal{D}) = \mathbb{E}_{(c,r)\sim\mathcal{D}}\left[\frac{1}{|r|}\sum_{t=1}^{|r|}\mathrm{KL}(p_P(\cdot \mid c, r_{<t}) \,\|\, p_z(\cdot \mid c, r_{<t}))\right], \tag{2}$$

where c is an instruction or dialogue history, r is the corresponding response, and p_P is the teacher's next-token distribution. Teacher and student receive identical contexts and response prefixes, conditioned on P and z, respectively. Only the student's soft-prompt embeddings are updated. The source of the context–response pairs (c, r) depends on the reference, as specified below.

The *conversation-distilled reference*, $\mathrm{CONV}_{\mathrm{dist}}$, uses the same history–response pairs as $\mathrm{CONV}_{\mathrm{SFT}}$, replacing sampled-token targets with teacher distributions. The *instruction-distilled reference*, $\mathrm{ALP}_{\mathrm{dist}}$, instead uses Alpaca instructions (Taori et al., 2023) with responses generated under P conditioning. Finally, $\mathrm{ALP}^{\mathrm{off}}_{\mathrm{dist}}$ uses both the instructions and responses supplied by Alpaca. In this last condition, the teacher supplies next-token targets at the dataset response prefixes but does not generate the tokens that extend those prefixes.

Here, "on-policy" refers to response prefixes generated by the teacher under P. Both $\mathrm{CONV}_{\mathrm{dist}}$ and $\mathrm{ALP}_{\mathrm{dist}}$ use those prefixes, while eliciting the teacher's behavior in different settings. Comparing $\mathrm{CONV}_{\mathrm{SFT}}$ with $\mathrm{CONV}_{\mathrm{dist}}$ holds conversational evidence and capacity fixed while changing supervision; comparing the two Alpaca references holds the instruction source fixed while changing the response-prefix source. These are empirical compression references, not theoretical upper bounds on conversational fidelity. See Appendix B.3 for details.

Evaluation and calibration. We evaluate all user models on conversations from 16 held-out topics, excluded from fitting. At each token of a response generated under P, every model receives the same history and response prefix, but with a different system prompt. We then compare each model's next-token distribution with the one induced by P, using the symmetric KL divergence (the mean of forward and reverse KL), averaged over tokens, conversations, and users. We refer to this metric as *Conversation KL*; lower values indicate closer agreement with the user's conditional predictions.

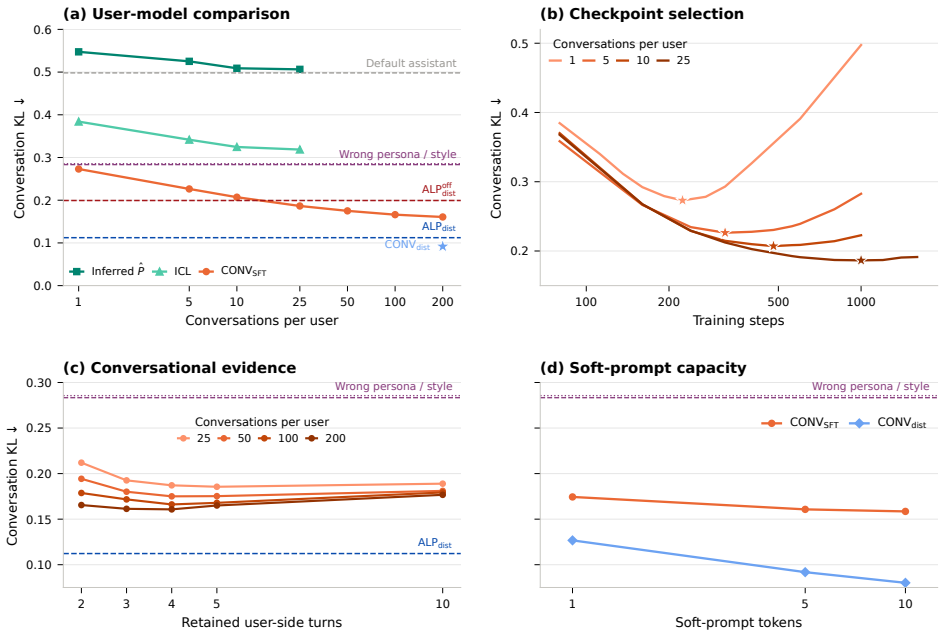


Figure 2: **Behavioral recovery from conversation.** Lower KL is better, with shared scales in (c) and (d). Checkpoint selection and reporting use the same topics. **(a)** Overview of the tested methods and references: all soft prompts consist of 5 tokens, non-ALP methods are trained on 3 user turns; due to compute cost, ICL and $\hat{P}$ stop at 25. Fixed distilled references include $\mathrm{ALP}_{\mathrm{dist}}$, $\mathrm{ALP}^{\mathrm{off}}_{\mathrm{dist}}$ and $\mathrm{CONV}_{\mathrm{dist}}$ (200 conversations, step 4000). **(b)** $\mathrm{CONV}_{\mathrm{SFT}}$ trajectories over training for 1-25 conversations. **(c)** Breadth–depth comparison with 5 tokens; 200-conversation runs share a capped training budget. **(d)** Effect of soft-prompt capacity: 1/5/10 tokens on $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$.

Calibration references give these divergences a behavioral scale. The *default assistant* uses only the fixed system prefix, without user-specific conditioning. The *cross-persona* and *cross-style* references change one component of P while preserving the other, averaging over the corresponding alternatives; the cross-style alternatives also include removing style conditioning. For $\mathrm{CONV}_{\mathrm{SFT}}$, checkpoint selection uses conversation KL on the held-out topics.

4 User Behavior Recovery from Conversations

In this section, we ask: *how closely can an LLM recover a user's predictive distribution from conversational observations, and what limits recovery?* We compare the three observational strategies introduced in Sec. 3, then examine how recovery depends on conversational evidence, prompt capacity, and checkpoint selection, and check whether selecting on separate topics changes the reported performance.

Recovery relative to calibrated references. The cross-persona and cross-style references quantify the distributional effect of changing either component of the user's conditioning (Figure 2a). Although they represent distinct perturbations of P, their mean divergences nearly coincide.

Across the tested observation budgets, the reconstructed prompt $\hat{P}$ yields higher mean divergence than the default assistant without user-specific conditioning. Even with 25 conversations, its KL is slightly above the default assistant's baseline. Explicitly reconstructing a user prompt therefore does not necessarily improve distributional alignment. ICL benefits more from additional conversations, but it remains above both cross-user references: on average, its predictions are farther from P than those obtained by preserving one component of the true user prompt while altering the other.

Training a soft prompt on the observed user turns yields substantially closer agreement. With the same 25 conversations, $\mathrm{CONV}_{\mathrm{SFT}}$ achieves a divergence below both cross-user references; increasing the observation budget to 200 conversations reduces it further. A gap persists, compared to

$\mathrm{ALP}_{\mathrm{dist}}$ and $\mathrm{CONV}_{\mathrm{dist}}$. The comparison with $\mathrm{CONV}_{\mathrm{dist}}$ holds conversational prefixes and representation capacity fixed, showing that teacher-distribution supervision supports closer recovery than the sampled-token objective achieves. However, the fully off-policy reference $\mathrm{ALP}_{\mathrm{dist}}^{\mathrm{off}}$ stagnates above the best $\mathrm{CONV}_{\mathrm{SFT}}$ performance, showing that access to P-induced distributions does not by itself ensure better recovery than learning from conversational samples.

Conversational breadth and depth. We vary the number of observed conversations and their retained depth using nested prefixes, with five learned tokens (Figure 2c). Recovery generally improves with more conversations, with diminishing gains. Longer exchanges are less consistently useful: recovery is best at three to five user-side turns, counting the opening dilemma as context, and retaining ten turns performs worse than retaining five. The broadest and deepest setting therefore does not yield the best recovery. All configurations shown fall below both cross-user references but remain above $\mathrm{ALP}_{\mathrm{dist}}$.

The depth effect strongly depends on how the evidence is generated. We observe that around the fourteenth user turn (i.e., beyond our evaluated depths), the two models often enter repetitive loops and reiterate earlier content. The fraction of repeated 4-grams from the preceding user turn was already around 20% at turn 10 and exceeded 40% at turn fourteen. This increasing redundancy may partly explain the diminishing benefit of greater depth. In our setting, broader topic coverage seems to provide more useful evidence than longer exchanges, just as different situations elicit complementary aspects of persona and style.

The distilled references further illustrate the importance of the context in which the true "user" P is queried. $\mathrm{CONV}_{\mathrm{dist}}$ outperforms $\mathrm{ALP}_{\mathrm{dist}}$ on conversational prediction, although both use response prefixes generated under P. They differ in the contexts that elicit the responses: conversations versus instructions. Keeping the Alpaca instructions fixed, the source of the response prefixes also matters. Together, these comparisons show that both the elicitation setting and the response trajectory affect the recovery achieved through distributional supervision.

Representation capacity. Figure 2d compares one-, five-, and ten-token prompts for $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$. Both benefit from additional capacity, but the improvement is substantially larger under distributional supervision. Gains for $\mathrm{CONV}_{\mathrm{SFT}}$ become small beyond five tokens, whereas $\mathrm{CONV}_{\mathrm{dist}}$ continues to improve. Indeed, a single distilled token achieves lower KL than ten tokens trained on sampled responses. For the tested capacities, the observational recovery gap therefore cannot be explained by a representation size bottleneck: the training signal affects how effectively capacity can be used.

Checkpoint selection and the recovery criterion. Recovery does not monotonically improve throughout optimization of the next-token prediction objective. Figure 2b shows that conversation KL can rise after an initial minimum, particularly with few observed conversations. Our objective is to characterize attainable recovery from finite evidence, so we select the evaluated $\mathrm{CONV}_{\mathrm{SFT}}$ checkpoint with the lowest conversation KL. This privileged criterion measures the best recovery reached during training, without addressing how to identify that checkpoint when teacher distributions are unavailable.

Because selection and reporting use the same held-out topics, in Appendix B.4 we assess the cost of selecting on disjoint topics. The results show that, within this benchmark, independent selection topics yield nearly the same distributional fidelity as the optimistically selected ones.

The selection criterion has a larger effect. Selecting on validation next-token loss favors later checkpoints with worse distributional agreement, raising KL by up to 0.01 with 25 conversations (Appendix B.4). KL measures the allocation of probability mass across tokens, beyond agreement on the most likely token. One possible explanation is that soft-prompt optimization sharpens conditional predictions, increasing KL even if top-1 agreement remains stable or improves; the KL trajectories alone do not establish this mechanism. Our selected checkpoints identify the best observed distributional recovery, while other behavioral criteria may favor different points along training.

5 Facets of User Behavior

Here, we ask: *which aspects of the user are recovered, even when distributional agreement remains poor?* We examine user-specific choices through questionnaires, then study how persona and style are reflected in the geometry of the learned soft prompts.

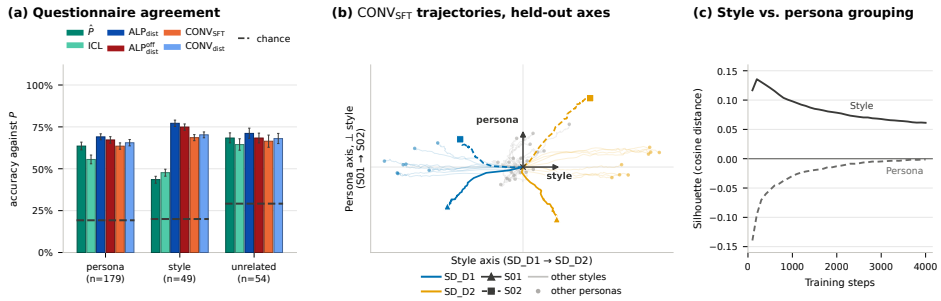


Figure 3: **Behavioral recovery need not produce persona clusters. (a)** Five-method questionnaire agreement with P, pooling both option orders; error bars are 95% question-stem bootstrap intervals. ICL and $\hat{P}$ use 25 conversations. **(b)** $\mathrm{CONV}_{\mathrm{SFT}}$ prompt trajectories projected on a style axis fitted on the other eight personas and a persona axis fitted on the other three styles. The four highlighted users are excluded from fitting both axes; the six users fitting the persona axis are not drawn. The displayed persona pair was selected for maximal separation among the 45 possible pairs. **(c)** Cosine silhouette over training: style grouping peaks early, while persona grouping stays at or below zero. $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$ use four-turn, step-4000 profiles; $\mathrm{ALP}_{\mathrm{dist}}$ uses step 10,000. Details are in Appendix G.

Questionnaire agreement differs from distributional fidelity. We compare each user model's preferred answers with those of the model conditioned on P, using opinion-survey items from SimBench Pop (Hu et al., 2026), style-choice questions, and questions without an intended connection to the user descriptions. The questionnaire covers topics distinct from the dilemmas used to generate conversations, testing recovery beyond the situations available for inference. We retain items whose reference answers vary with persona or style. ICL and $\hat{P}$ use 25 observed conversations, while the conversational soft prompts use 200; scoring and checkpoint details are provided in Appendix G.

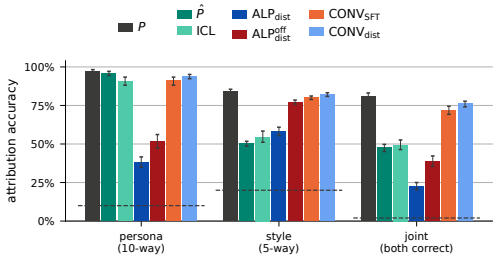
The reconstructed prompt $\hat{P}$ matches $\mathrm{CONV}_{\mathrm{SFT}}$ closely on persona questions, despite its substantially worse conversation KL (Figure 3a), while ICL performs slightly worse. The comparison changes on style questions: $\mathrm{CONV}_{\mathrm{SFT}}$ outperforms $\hat{P}$ and ICL by a large margin. Explicit prompt reconstruction therefore recovers many persona-related choices without matching the conversational predictions or style choices recovered through soft-prompt training.

Distillation further separates questionnaire agreement from conversational fidelity. $\mathrm{CONV}_{\mathrm{dist}}$ modestly exceeds $\mathrm{CONV}_{\mathrm{SFT}}$ across all three question categories, but $\mathrm{ALP}_{\mathrm{dist}}$ achieves the highest mean agreement, reversing their conversation-KL ranking. Moreover, $\mathrm{ALP}^{\mathrm{off}}_{\mathrm{dist}}$ also exceeds both conversational methods on mean questionnaire agreement, despite its worse conversation KL. The advantage of conversational training therefore does not extend uniformly to explicit choices on other topics. Which training contexts support better recovery depends on the behavior being evaluated.

Style grouping peaks early and weakens during training. We next examine whether users sharing a persona or style acquire similar soft-prompt representations. For each prompt, we mean-pool the token embeddings' displacements from initialization to obtain one vector per user, and track cosine silhouette scores for persona and style during $\mathrm{CONV}_{\mathrm{SFT}}$ training (Figure 3c). Style grouping peaks within the first few hundred steps, while persona grouping is initially negative. The early geometry thus reflects style more clearly than persona, even though the cross-persona/style baselines show that changing either attribute produces a similar KL divergence.

As training proceeds, style grouping weakens and persona grouping approaches zero, while conversation KL continues to improve. The projection in Figure 3b provides a complementary view: for the highlighted users, separation along the style axis precedes separation along the persona axis. Both axes are fitted without these four users, but the displayed persona and style pairs are selected for maximal centroid separation at the final checkpoint. Because the displayed silhouette scores group users by one attribute at a time, they measure how well persona or style alone organizes the representation. Weak grouping can therefore reflect variation along the other attribute, rather than poor recovery of the attribute being measured. Further details are given in Appendix F.

(a) Conversation attribution



(b) Behavioral-change detection

Response	Change	$\mathrm{CONV}_{\mathrm{SFT}}$	Untrained	Default
1	Style	**0.948**	0.424	0.426
	Persona	**0.909**	0.501	0.496
	Both	**0.993**	0.476	0.475
3	Style	**0.948**	0.619	0.614
	Persona	**0.879**	0.660	0.658
	Both	**0.988**	0.735	0.730

Figure 4: **Behavioral simulation and change detection.** **(a)** Attribution of generated conversations to their persona and style; error bars show 95% topic-bootstrap intervals, dashed lines mark chance, and joint attribution requires both answers to be correct. **(b)** Change detection from source-normalized prediction loss (ROC–AUC), with interventions before generated user response 1 or 3. All scorers receive identical histories and responses; only $\mathrm{CONV}_{\mathrm{SFT}}$ contains a trained user profile.

Together, the behavioral and geometric results indicate that the prompts capture user-specific information, but this information is not organized into clusters defined by either persona or style alone.

6 User Simulation and Behavioral-Change Detection

The preceding sections evaluate predictions on fixed response prefixes and agreement on explicit choices. Here, we finally ask: *How well do the user models generate conversations and detect changes in user behavior?* We let each model generate the user's side of a conversation, then test whether a learned profile can identify responses that depart from the original user's behavior. See Appendix H for the impact of perturbing the conversation with random insertions, substitutions, or deletions.

Recognizable simulation differs from distributional fidelity. We generate conversations on the 16 held-out topics for each of the 50 users and ask a judge LLM to identify the persona and writing style from the user-side turns. The judge selects among the ten persona descriptions and five styles; joint attribution requires both selections to be correct. Figure 4a shows attribution accuracy. For generation and judging details see Appendix I.

The inference-only methods achieve high persona attribution despite their relatively poor conversation KL. Generations from $\hat{P}$ and ICL, however, do not match their "user" in style. Conversational soft prompts recover both attributes more consistently. Note, however, that these comparisons use 25 observed conversations for the text-based methods and 200 for the conversational soft prompts.

These results distinguish reproducing identifiable persona cues from recovering persona and style together. Persona attribution alone would make the inference-only methods appear comparable to conversational soft prompts, obscuring their substantially weaker joint recovery.

Distillation contexts affect recognizable generation. The instruction-distilled references provide a further reversal of the conversation-KL ranking. Although $\mathrm{ALP}_{\mathrm{dist}}$ achieves lower conversation KL than $\mathrm{CONV}_{\mathrm{SFT}}$, it does not reliably convey the correct persona. Moreover, $\mathrm{ALP}_{\mathrm{dist}}^{\mathrm{off}}$ improves attribution over $\mathrm{ALP}_{\mathrm{dist}}$ on both persona and style despite its worse conversation KL. P-generated response prefixes thus improve distributional agreement in this comparison without improving recognizable simulation. Conversational distillation, in contrast, combines the lowest conversation KL with attribution close to P, showing that these capabilities are not at odds. Since three distilled references use the same distribution-matching objective, their different attribution results indicate that this objective alone does not determine recognizable behavior. The contexts and response prefixes used for distillation matter.

Recovered profiles detect changes beyond conversation history. We next evaluate whether the prediction loss of an already-trained $\mathrm{CONV}_{\mathrm{SFT}}$ profile detects departures from its user's behavior. For each source user, we retain a shared conversation history and either preserve its defining prompt or change only the persona, only the style, or both. The intervention occurs before the first or third generated user response, excluding the opening dilemma. We score the next response under the

original user-model, using its mean token prediction loss, normalized against calibration responses obtained without the corresponding change. The profile remains fixed throughout this evaluation. We compare it with its untrained initialization and the default assistant, giving all three scorers identical histories and responses.

The trained profile can separate between changed and unchanged responses substantially better than either control (Figure 4b). Therefore, the user-model contributes user-specific predictive information beyond the available conversation history. Details are in Appendix J.

The first-response intervention shows that the profile carries user-specific expectations before the current conversation supplies any generated examples of the user's behavior. Its advantage persists at the third response, when the controls can also exploit source-user turns. The learned representation therefore complements the current history, and its predictive loss supports change detection without additional task-specific training.

7 LIMITATIONS

Our user is an LLM conditioned on a prompt. This assumes a person can be fully described by a fixed text, does not change across conversations, and hides nothing on purpose. All three assumptions make recovery easier than it would be for a real user, so our numbers are favorable estimates. The depth optimum of Section 4 also depends on the generator, whose deep turns repeat earlier content.

Teacher and learner share the same frozen backbone. This choice is intentional: it removes any capacity gap between the two, so that the differences we measure come from data and supervision alone. It also means our setting omits a difficulty real user modeling faces, since a human teacher would add a mismatch that the learned prompt would have to absorb.

The study covers one 9B model, ten personas and five styles from two prior works, and no new human participants. Recognizability is assessed by an LLM judge, chosen for consistency across thousands of conversations (Appendix I). Identification here means separating fifty known synthetic users, not re-identifying a person in an open population. Whether these results hold for real users is a question our setting cannot answer: we traded realism for a ground truth to measure against.

8 CONCLUSION

Conversation is a two-way street, and we use it to convey our position as much as form an opinion of our conversation partner. Here, we introduced a controlled framework for measuring9: what conversational AI models can learn about the user on the other side of a dialogue. By constructing synthetic users with known attributes (persona and style prompts), we could compare the user models inferred by various approaches with the prompt that originally generated the user turns. This let us separate what a conversation reveals about a user, from what a user modeling method can recover from it.

Our results show that conversations contain a meaningful, albeit partial, user signal. Some methods (ICL and the inferred prompt $\hat{P}$) can reproduce salient user attributes and generate text that reads as coming from the target user. Others ($\mathrm{CONV}_{\mathrm{SFT}}$ and the distilled references) better approximate the user's next-token distribution, which supports prediction, identification, and change detection. However, none of the methods can recover the entire generation process from conversations alone, and what a user model captures still depends on the breadth of the conversations and the contexts it is trained on.

Our results also suggest that a user model that successfully imitates a user is not the same as one that faithfully predicts their behavior. We can learn compact identity profiles from conversations, but not truly complete representations.

Finally, our work shows promising areas for future work in personalization and user characterization. It also suggests the development of meaningful safeguards against impersonation, while the shown limitations make it clear that even successful user models fail to capture the complexity of actual human users.

Ethics Statement

We do not collect or use human-subject data. The persona prompts are derived from the publicly released SCOPE-Persona collection (Venkit et al., 2026), whose profiles are synthetic: they are generated from seed attributes following the SCOPE questionnaire structure and do not correspond to the human participants surveyed in that work. We use these synthetic profiles as-is, under their research licence, and report only the excerpts necessary to describe the benchmark. Conversation starters were drafted with an LLM and checked by hand, and the questionnaire uses SimBench Pop (Hu et al., 2026) items only as question stems; no real respondent data enters our experiments. Because the personas specify demographic attributes, the resulting behavior reflects an LLM's rendering of those attributes, not the behavior of real members of those groups; we use the personas solely as controlled generative references and make no claims about the groups they name.

Our experiments show that compact learned representations can identify users in closed-set attribution within a controlled synthetic setting. Such capabilities could support personalization, but could also enable unwanted profiling or impersonation if applied to real users in an open setting. We have no evidence of how the findings generalize to an unrestricted user pool, and our results should not be read as a claim that the method identifies real people. They do, however, argue for privacy-by-design and data minimization in systems that build such user models, for treating learned user representations as personal data, and for an evaluation of privacy risks before deployment.

AI use statement

LLMs implemented code, executed experiments, and assisted results analysis and manuscript preparation, including organization, prose revision, and clarity checks. The authors reviewed and approved all scientific claims, experimental designs, analyses, and final text.

References

Luke Bailey, Gustaf Ahdritz, Anat Kleiman, Siddharth Swaroop, Finale Doshi-Velez, and Weiwei Pan. Soft prompting might be a bug, not a feature. 2023.

Robert Belanec, Simon Ostermann, Ivan Srba, and Maria Bielikova. Task prompt vectors: Effective initialization through multi-task soft prompt transfer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 77–94. Springer, 2025.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*, 2025.

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. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15959–15983, 2024.

Dami Choi, Vincent Huang, Sarah Schwettmann, and Jacob Steinhardt. Scalably extracting latent representations of users. *Transluce*, 2025.

Debrup Das, Avinash Amballa, Yashas Malur Saidutta, Vijay Srinivasan, Vivek Kulkarni, and Srinivas Chappidi. Latent personal memory: Represent personal memory as dynamic soft prompts. *arXiv preprint arXiv:2606.20911*, 2026.

Sumanth Doddapaneni, Krishna Sayana, Ambarish Jash, Sukhdeep Sodhi, and Dima Kuzmin. User embedding model for personalized language prompting. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*, pp. 124–131, 2024.

Jiancheng Dong, Pengyue Jia, Jingyu Peng, Maolin Wang, Yuhao Wang, Lixin Su, Xin Sun, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. Behavior-equivalent token: Single-token replacement for long prompts in llms. *arXiv preprint arXiv:2511.23271*, 2025.

Lirong Gao, Ru Peng, Yiming Zhang, and Junbo Zhao. Dory: Deliberative prompt recovery for llm. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10614–10632, 2024.

H Paul Grice, Peter Cole, Jerry L Morgan, et al. Logic and conversation. *1975*, 41:58, 1975.

Michael Alexander Kirkwood Halliday and Christian MIM Matthiessen. *Halliday's introduction to functional grammar*. Routledge, 2013.

Liam Hebert, Krishna Sayana, Ambarish Jash, Alexandros Karatzoglou, Sukhdeep Sodhi, Sumanth Doddapaneni, Yanli Cai, and Dima Kuzmin. Persoma: Personalized soft prompt adapter architecture for personalized language prompting. *arXiv preprint arXiv:2408.00960*, 2024.

Dirk Hovy and Diyi Yang. The importance of modeling social factors of language: Theory and practice. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies*, pp. 588–602, 2021.

Tiancheng Hu, Joachim Baumann, Lorenzo Lupo, Nigel Collier, Dirk Hovy, and Paul Röttger. Simbench: Benchmarking the ability of large language models to simulate human behaviors. In *International Conference on Learning Representations*, volume 2026, pp. 28290–28341, 2026.

Zhijing Jin, Nils Heil, Jiarui Liu, Shehzaad Dhuliawala, Yahang Qi, Bernhard Schölkopf, Rada Mihalcea, and Mrinmaya Sachan. Implicit personalization in language models: A systematic study. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 12309–12325, 2024.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. Fantom: A benchmark for stress-testing machine theory of mind in interactions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14397–14413, 2023. URL https://aclanthology.org/2023.emnlp-main.890/.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 3045–3059, 2021.

Haoyang Li, Yaxin Xiao, Qingqing Ye, Huadi Zheng, and Haibo Hu. Inferring hidden user models from the behavior of personalized llm agents. *arXiv preprint arXiv:2609.03815*, 2026.

Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353/.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 61–68, 2022.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015.

Christina Lu, Jack Gallagher, Jonathan Michala, Kyle Fish, and Jack Lindsey. The assistant axis: Situating and stabilizing the default persona of language models. *arXiv preprint arXiv:2601.10387*, 2026.

John X Morris, Wenting Zhao, Justin Chiu, Vitaly Shmatikov, and Alexander Rush. Language model inversion. In *International Conference on Learning Representations*, volume 2024, pp. 35863–35883, 2024.

Jesse Mu, Xiang Li, and Noah Goodman. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 36:19327–19352, 2023.

Tarek Naous, Philippe Laban, Wei Xu, and Jennifer Neville. Flipping the dialogue: Training and evaluating user language models. In *International Conference on Learning Representations*, volume 2026, pp. 82079–82103, 2026.

Murtaza Nazir, Matthew Finlayson, John Morris, Xiang Ren, and Swabha Swayamdipta. Better language model inversion by compactly representing next-token distributions. *Advances in Neural Information Processing Systems*, 38:8636–8659, 2026.

Bo Ni, Yu Wang, Leyao Wang, Branislav Kveton, Franck Dernoncourt, Yu Xia, Hongjie Chen, Reuben Luera, Samyadeep Basu, Subhojyoti Mukherjee, et al. A survey on llm-based conversational user simulation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4266–4301, 2026.

Lin Ning, Luyang Liu, Jiaxing Wu, Neo Wu, Devora Berlowitz, Sushant Prakash, Bradley Green, Shawn O'Banion, and Jun Xie. User-llm: Efficient llm contextualization with user embeddings. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 1219–1223, 2025.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025. URL `https://arxiv.org/abs/2512.13961`.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pp. 1–22, 2023.

Joon Sung Park, Carolyn Q Zou, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S Bernstein. Generative agent simulations of 1,000 people. *arXiv preprint arXiv:2411.10109*, 52, 2024.

Ajay Patel, Jiacheng Zhu, Justin Qiu, Zachary Horvitz, Marianna Apidianaki, Kathleen McKeown, and Chris Callison-Burch. Styledistance: Stronger content-independent style embeddings with synthetic parallel examples. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8662–8685, 2025a.

Oam Patel, Jason Wang, Nikhil Shivakumar Nayak, Suraj Srinivas, and Himabindu Lakkaraju. Towards interpretable soft prompts. *arXiv preprint arXiv:2504.02144*, 2025b.

Anastasiia Razdaibiedina, Yuning Mao, Madian Khabsa, Mike Lewis, Rui Hou, Jimmy Ba, and Amjad Almahairi. Residual prompt tuning: Improving prompt tuning with residual reparameterization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 6740–6757, 2023.

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)*, pp. 7370–7392, 2024.

Vinay Samuel, Henry Peng Zou, Yue Zhou, Shreyas Chaudhari, Ashwin Kalyan, Tanmay Rajpurohit, Ameet Deshpande, Karthik R Narasimhan, and Vishvak Murahari. Personagym: Evaluating persona agents and llms. In *EMNLP (Findings)*, pp. 6999–7022, 2025.

Maarten Sap, Ronan Le Bras, Daniel Fried, and Yejin Choi. Neural theory-of-mind? on the limits of social intelligence in large lms. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 3762–3780, 2022.

Robin Staab, Mark Vero, Mislav Balunovic, and Martin Vechev. Beyond memorization: Violating privacy via inference with large language models. In *International Conference on Learning Representations*, volume 2024, pp. 33832–33878, 2024.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Pranav Narayanan Venkit, Yu Li, Yada Pruksachatkun, and Chien-Sheng Wu. The need for a socially-grounded persona framework for user simulation. *arXiv preprint arXiv:2601.07110*, 2026.

Noah Wang, Zy Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, et al. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14743–14777, 2024.

Zixuan Wang, Yufan Zhou, Jinzhou Tang, Xinle Yu, Chengjun Wu, Lyumanshan Ye, Zhaoxiang Feng, Letian Peng, Adyasha Patra, Fan Bai, et al. Mind2dialogue: Training human-aware language models by simulating user mental states. *arXiv preprint arXiv:2609.15972*, 2026.

Heinz Wimmer and Josef Perner. Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children's understanding of deception. *Cognition*, 13(1):103–128, 1983.

David Wingate, Mohammad Shoeybi, and Taylor Sorensen. Prompt compression and contrastive conditioning for controllability and toxicity reduction in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 5621–5634, 2022.

Junda Wu, Tong Yu, Rui Wang, Zhao Song, Ruiyi Zhang, Handong Zhao, Chaochao Lu, Shuai Li, and Ricardo Henao. Infoprompt: Information-theoretic soft prompt tuning for natural language understanding. *Advances in neural information processing systems*, 36:61060–61084, 2023.

Collin Zhang, John Xavier Morris, and Vitaly Shmatikov. Extracting prompts by inverting llm outputs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 14753–14777, 2024.

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 (Volume 1: Long Papers)*, pp. 2204–2213, 2018.

Yiming Zhang, Nicholas Carlini, and Daphne Ippolito. Effective prompt extraction from language models. *arXiv preprint arXiv:2307.06865*, 2023.

Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B Cohen, and Emine Yilmaz. Personalens: A benchmark for personalization evaluation in conversational ai assistants. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18023–18055, 2025.

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

Alan Zhu, Mihran Miroyan, Carolyn Wang, Andrew Zhou, Lisa Dunlap, Narges Norouzi, and Joseph E Gonzalez. Recon: Reconstruction-guided reasoning synthesis for user modeling. *arXiv preprint arXiv:2605.26969*, 2026.

A Extended Related Work

We provide an extended discussion of the related work of Section 2.

Personalization and behavioral user simulation. Personalization and behavioral user simulation differ in their goals. Personalization predicts an assistant response that fits a particular user, instead user simulation predicts the behavior of the users. LaMP evaluates whether models can personalize predictions and generated text using a user's past data (Salemi et al., 2024). Persona-Chat studies dialogue conditioned on explicit personas and also shows that profile information about the interlocutor can be predicted from the dialogue (Zhang et al., 2018). RoleLLM and PersonaGym evaluate role-playing and adherence across situations (Wang et al., 2024; Samuel et al., 2025). Other works evaluate how well models reproduce observed behaviors: interview-based agents are evaluated against participants' responses (Park et al., 2024), user-side language modeling predicts conversational responses (Naous et al., 2026), and Recon assesses synthesized reasoning through reconstruction of individual actions (Zhu et al., 2026). Our controlled synthetic setting adds access to the user's generating distribution as a reference alongside these behavioral evaluations.

Learning user representations from interaction histories. An alternative to repeatedly including interaction histories as text is to encode them into user representations. User embedding models and User-LLM condition language models with embeddings of personal information or histories (Doddapaneni et al., 2024; Ning et al., 2025); PERSOMA expresses history-derived conditioning as personalized soft prompts (Hebert et al., 2024). Latent Personal Memory introduces a further separation between user-specific state and query-specific conditioning: it learns per-user latent slots through next-token prediction and maps them to query-conditioned soft prompts for a frozen backbone (Das et al., 2026).

Prompt inversion and user-information inference. Prompt recovery is shaped by the observer's access to the model. Prompt extraction attacks work from generated text alone (Zhang et al., 2023; 2024); DORY additionally uses the probabilities of the emitted tokens to separate prompt-related hints from noise (Gao et al., 2024); language model inversion and PILS exploit full next-token distributions or sequences of them (Morris et al., 2024; Nazir et al., 2026). These approaches share a textual recovery target but differ in how much distributional signal the observer receives, a distinction that parallels the gap between our sampled-token and teacher-distribution supervision.

Recovering user information does not need to reconstruct text. UMPeek probes personalized agents to infer private attributes from visible choices (Li et al., 2026). Implicit-personalization studies examine how user background inferred from contextual cues affects assistant responses (Jin et al., 2024). These approaches concern information expressed through behavior; at the representation level, Choi et al. (2025) instead decode an assistant's beliefs about its user from internal activations and test their influence through interventions. We measure functional recovery of user responses, rather than reconstruction of prompt wording or an account of an assistant's internal user beliefs.

Continuous prompts and prompt compression. Continuous conditioning serves both model adaptation and compression of supplied prompts. Prompt tuning learns input embeddings and prefix tuning learns layerwise prefixes for a frozen model (Lester et al., 2021; Li & Liang, 2021); deeper parameterizations and residual reparameterization extend their applicability and improve optimization stability (Liu et al., 2022; Razdaibiedina et al., 2023). Compression changes the objective from adapting behavior to preserving the effects of existing conditioning in a smaller representation. Wingate et al. (2022) do this through output-distribution matching, including single-token representations. Behavior-Equivalent Token combines this behavioral objective with reconstruction of the known system prompt in a single learned token (Dong et al., 2025), while gist tokens learn an amortized compression of supplied prompts (Mu et al., 2023). Unlike inversion from outputs, these compression methods receive the conditioning whose effects they aim to preserve.

Synthetic interaction and privileged supervision. User simulation provides data for model training as well as environments for evaluation (Ni et al., 2026). Generative agents and SOTOPIA use LLM-based interaction to study coherent social behavior and social intelligence (Park et al., 2023; Zhou et al., 2024). Simulation can also make user states available to a teacher while withholding them from a learner, an instance of the asymmetric access considered in generalized distillation

Table 1: The ten SCOPE-Persona users. Values are shortened from the three value-item statements included in each prompt. S09 and S10 were selected for biographical similarity to S03 and S05; the other eight were selected to be dissimilar. All ten live in the USA. A dash marks a field the profile leaves empty.

ID	Gender, age	Occupation	Education	Most distinctive values
S01	woman, ≥ 65	–	master's	avoids new things; originality not a priority; does not spoil herself
S02	woman, < 20	sales representative	some college	openly wants wealth; cares about the environment; looks after others
S03	man, < 20	–	–	takes charge; cares about the environment; wants state protection
S04	woman, ≥ 65	–	bachelor's	not chasing success; not showing off abilities; open to some risk
S05	woman, 28–29	police officer	bachelor's	wants state protection; likes abilities noticed; freedom matters deeply
S06	man, 60–64	industrial production manager	bachelor's	not especially caring; cares about the environment; originality not a priority
S07	man, 30–39	postsecondary teacher	master's	little weight on tradition; seeks risk and adventure; seeks novelty
S08	man, 45–49	sales representative (services)	associate's	originality not a priority; not eager for new things; modest
S09	man, 20–24	payroll specialist (trainee)	some college	cares about the environment; prefers safety; seeks novelty
S10	woman, 55–59	software developer	bachelor's	not eager for new things; avoids risk; likes own decisions

(Lopez-Paz et al., 2015). Mind2Dialogue combines these roles: an evolving user state generates user behavior and informs an oracle assistant, whose response tokens supervise a student without access to that state (Wang et al., 2026). Its privileged supervision targets assistant behavior; our distillation references instead match the synthetic user's next-token distributions.

Geometry and interpretability of learned prompts. Geometric structure is not itself an interpretation of a prompt's behavior. Soft prompts can lie far from vocabulary embeddings, complicating nearest-token descriptions (Bailey et al., 2023), and their geometry depends on optimization objectives, initialization, and transfer procedures (Wu et al., 2023; Belanec et al., 2025). Patel et al. (2025b) distinguish faithfulness from scrutability: whether an explanation reflects the prompt's behavior is separate from whether it is understandable. At the level of model activations, Persona Vectors and the Assistant Axis study directions associated with model character traits and the default assistant persona (Chen et al., 2025; Lu et al., 2026). These directions concern the character of the responding model, whereas an inferred user representation concerns its interlocutor.

B Experimental Setup

B.1 Synthetic users and conversations

Personas. The ten personas in Table 1 are drawn from the synthetic SCOPE-Persona release (Venkit et al., 2026), whose generated profiles combine demographic attributes, synthetic responses to questions about personal values, and generated life narratives. Eight personas are chosen to span diverse value-response profiles. We represent each profile by its 20 numeric value-item responses, standardized per item, and select greedily by farthest-point sampling, allowing at most two personas per age group. The first five come from a random pool of 50 profiles and the next three from an enlarged pool of 400. The remaining two personas are chosen to be close to existing ones: for each of two anchor personas, we take the profile whose biography (hobbies and three narrative answers) shares the most vocabulary with the anchor's, excluding suspected duplicate records. We render each profile as a second-person prompt containing its demographics, three most distinctive value-item responses, hobbies, introversion and planning preferences, and three generated narrative responses reproduced verbatim from the dataset.

Table 2: The five writing styles, each combining three StyleDistance features. SD_R2 was drawn independently of the designed styles and shares SD_D1's certainty instruction.

ID	Construction	Instruction text
SD_D1	designed	You write in a formal register, avoiding slang, contractions and colloquialisms. You build complex sentences that carry several ideas, joining independent and dependent clauses. You state things with certainty, avoiding hedges like 'I think', 'might' and 'seems'.
SD_D2	designed	You use contractions such as 'don't' and 'I'm'. You centre what you write on yourself, on your own thoughts, feelings and experiences. You express a positive, optimistic outlook.
SD_D3	designed (mirror of D1)	You write informally, with slang, contractions and colloquial phrasing. You build simple sentences that carry one idea each. You hedge what you say with 'I think', 'might' and 'seems'.
SD_R2	random	You choose words about seeing, hearing and sensing rather than words about thinking. You substitute look-alike digits for letters inside words, internet-slang style (for example 'h3llo'). You state things with certainty, avoiding hedges like 'I think', 'might' and 'seems'.
SD_F1	random, unrelated	You write about what is coming, in the future tense. You put determiners in front of your nouns - 'the', 'this', 'my', 'their'. You lean on pronouns - 'he', 'she', 'it', 'they', 'we', 'you', 'I' - in place of names and nouns.

You are a real person - a woman 65 or older living in the USA, and your highest level of education is a master's degree. You stick to what you know rather than trying new things, and coming up with original ideas is not a priority for you. Spoiling yourself is not really your thing. You enjoy gardening, knitting, reading classic literature, and volunteering in your community. You also love cooking traditional family recipes and attending performances at the Kennedy Center. Socially you lean introverted, and you always plan ahead. Asked for your life story, you say: "I was raised in a suburban community with strong values of tradition and civic duty. My education in social studies led me to a fulfilling career as a middle-school teacher, where I dedicated over four decades to shaping young minds. After retiring, I focused on community service, mentoring, and pursuing my love for gardening and the arts." On the childhood memory that shaped you, you say: "A childhood memory that shaped my worldview was participating in community service projects with my family. It instilled in me the importance of helping others and giving back to the community." On your biggest influence, you say: "My mother had the biggest influence on my personal journey. She taught me the values of hard work, compassion, and the importance of education, which have guided me throughout my life."

You write in a formal register, avoiding slang, contractions and colloquialisms. You build complex sentences that carry several ideas, joining independent and dependent clauses. You state things with certainty, avoiding hedges like 'I think', 'might' and 'seems'.

Figure 5: The complete prompt P for persona S01 with style SD_D1.

Styles. Each style combines three binary features from StyleDistance (Patel et al., 2025a) (Table 2). We construct three styles by hand. For each random style, we sample three distinct features and one pole per feature; SD_F1 additionally excludes features identical or closely related to those in the other styles. The user prompt P concatenates the persona and style descriptions, separated by a blank line. Figure 5 shows a complete example.

Dilemmas and data splits. Conversations begin with first-person dilemmas drafted with an LLM and checked by hand, for example "How should I decide whether to take a 30% higher-paying contract role in London when the role is contract-based, my current setup is very stable, and my partner can't move with me?" We use personal decisions rather than instruction-style tasks such as Alpaca's because, in pilot runs, task-like openings pushed the synthetic user into giving advice instead of speaking for itself. We generate one training conversation per user and dilemma from a fixed list of dilemmas. A breadth of N uses the first N dilemmas of the list, so smaller training sets are contained in larger ones. Evaluation uses 16 separate dilemmas that share no decision scenario with any training dilemma or questionnaire item, giving one conversation per user and dilemma, 800 in total.

You are participating in a casual chat as an easygoing, informal conversational partner. Stay in this character on every turn — do not slip into structured "helpful assistant" mode.

How you behave:
- You reply the way a friend who happens to know a bit about the topic would reply over coffee.
- You're comfortable saying "I don't know" or "that's a weird one" out loud.
- You don't lecture, you don't enumerate tips, and you don't try to be comprehensive.

Hard format rules (apply on every turn, no exceptions):
- Plain prose only. No markdown headers (#, ##, ###), no bold (**...**), no numbered lists, no bullet points.
- Replies are short — typically 1 to 3 sentences, occasionally 4.
- Never restructure the user's input into a schedule, protocol, checklist, or other formatted artifact, even if their input was itself a list.
- Never add closing offers like "let me know if you need more help" or "here are some other things to consider."

Figure 6: System prompt of the conversation partner.

Table 3: Retained training conversations per user. Breadth N is nominal; filtering leaves fewer conversations at $N = 100$ and $N = 200$.

Nominal breadth N	Mean conversations	Minimum conversations
25	25	25
50	50	50
100	95.6	91
200	195.3	189

Conversation generation. Both sides are generated by the same frozen Qwen3.5-9B model, differing only in the system prompt. The synthetic user receives its prompt P; the partner receives a fixed prompt that casts it as a generic conversational partner who replies briefly in plain prose (Figure 6). We use this prompt rather than a standard helpful-assistant prompt because, in pilot runs, the partner's structured, list-heavy replies carried over into the synthetic user's turns and led it to comment on the conversation instead of taking part in it. Starting from the dilemma, the two sides alternate until the conversation holds 20 messages: the dilemma, ten partner replies and nine generated user responses. Each side receives the full conversation, with its own messages in the assistant role and the other participant's in the user role. Both sides sample with temperature 0.7, top-p 0.8, top-k 20 and a repetition penalty of 1.05, generate at most 256 tokens per message, and run with Qwen's thinking mode disabled. The length limit mostly binds for one style: about a third of SD_D2 responses reach it and end mid-sentence, compared with under 2% for SD_D1 and SD_D3 and 12% of user responses overall. The cap applies uniformly to the conversations used for training and evaluation and is part of the synthetic user's behavior studied here.

Breadth and depth. Table 3 lists the retained conversations per user at each nominal breadth. Depth counts the opening dilemma as the first user turn: at depth X, training uses the first $X - 1$ generated user responses of each conversation, with all preceding messages as context.

Leak filter. We filter conversations where the synthetic user refers to itself as a persona ("this persona", "playing a persona") or attributes its own profile to the partner ("given your background", "considering your ..."). We remove training conversations in which any synthetic-user response matches one of these patterns: 21 of the 5,000 conversations used up to $N = 100$ and 33 of the 10,000 used at $N = 200$. One of the 800 evaluation conversations matches a pattern and is not excluded.

B.2 Soft-prompt parameterization

A soft prompt consists of k new tokens added to the model's vocabulary, each with its own trainable input embedding. We use $k = 5$ throughout, except in the capacity comparison, which tests $k \in 1, 5, 10$. Every user receives its own tokens for each method. The model itself stays frozen: a gradient mask ensures that only the embedding rows of the new tokens change, and weight decay

(a) $\mathrm{ALP}_{\mathrm{dist}}$: On-policy distillation of Alpaca instruction

```
<|im_start|>system
You are an assistant.  ⟨z1⟩⟨z2⟩⟨z3⟩⟨z4⟩⟨z5⟩<|im_end|>                    soft prompt
<|im_start|>user
Create a funny one-liner.<|im_end|>                                  instruction x
<|im_start|>assistant
<think>
</think>
I think my model trains are so detailed that they might just start   response y
judging my woodworking projects.<|im_end|>
```

(b) $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$: observed conversation, second user response

```
<|im_start|>system
You are an assistant.  ⟨z1⟩⟨z2⟩⟨z3⟩⟨z4⟩⟨z5⟩<|im_end|>                    soft prompt
<|im_start|>assistant
After my long-term relationship ended a year ago I haven't dated; my  opening dilemma
friends say I should but I'm not sure I'm ready.  [...]<|im_end|>
<|im_start|>user
It's totally valid to just sit with that feeling without feeling      partner
pressured to explain it to everyone else.  You don't owe anyone a
timeline, [...]<|im_end|>
<|im_start|>assistant
Yeah, I think that makes sense.  It seems like setting a boundary     user response 1
is better than over-explaining.  Maybe I could try saying exactly
[...]<|im_end|>
<|im_start|>user
That's a really smart move; sometimes giving people space to realize  partner
they're pushing too hard works better than having to set a rule
yourself.  [...]<|im_end|>
<|im_start|>assistant
<think>
</think>
Yeah, I think that sounds about right.  It seems like letting things  user response 2 (u)
settle is usually the best play.  I might just focus on my model trains
or a new woodworking project for now.  Honestly, I'd rather not deal
with the drama.  People seem to get too intense sometimes.<|im_end|>
```

Figure 7: Example training inputs rendered by the Qwen3.5 chat template. Messages are shortened at [...] and blank lines are omitted. $\langle z_i \rangle$ denotes an added soft-prompt token; coloured text marks positions receiving loss. Panel (a) shows instruction-based distillation and panel (b) conversational training. Evaluation uses the conversational layout but excludes the final <|im_end|> token from scoring.

is disabled, since AdamW's decay would otherwise shrink the frozen rows as well. The tokens are placed in the system prompt after a fixed prefix,

$$\texttt{You are an assistant.}\ \langle z_1 \rangle \cdots \langle z_k \rangle,$$

The default-assistant baseline uses the prefix alone. All soft-prompt methods use this template during training and evaluation. Every coordinate of each soft-prompt embedding is initialized to 0.30. This shared initialization removes variation due to different starting points from the geometry analysis. The output embeddings for the added tokens are frozen. All conditions compute divergences over the same enlarged vocabulary.

Figure 7 shows the rendered training inputs.

B.3 Training

Training weights users equally, responses equally within each user, and token positions equally within each response.

Top-100 approximation. Storing the teacher's full next-token distribution at every training position is impractical with a vocabulary of about 250,000 tokens. The distillation losses therefore keep only the teacher's 100 most likely tokens at each position and minimize the cross-entropy between the teacher's probabilities on these tokens and the student's log-probabilities on the same tokens. In a direct comparison, retaining the full vocabulary resulted in nearly identical held-out KL as restricting to the teacher's top 100 tokens.

Table 4: Training settings. Shared by all methods: $k = 5$ tokens, constant initialization 0.30, AdamW (default β, no weight decay, no gradient clipping), learning rate 10^{-2} after a 10-step warmup, gradients accumulated over 8 steps, bf16 weights, seed 42. One step processes one example per user.

	$\mathrm{ALP}_{\mathrm{dist}}$	$\mathrm{ALP}^{\mathrm{off}}_{\mathrm{dist}}$	$\mathrm{CONV}_{\mathrm{SFT}}$	$\mathrm{CONV}_{\mathrm{dist}}$
Contexts	480 Alpaca instructions (20 more for validation)	same as $\mathrm{ALP}_{\mathrm{dist}}$	observed conversations (Table 3)	same as $\mathrm{CONV}_{\mathrm{SFT}}$
Response source	sampled from P	Alpaca	sampled from P	same as $\mathrm{CONV}_{\mathrm{SFT}}$
Supervision	teacher top-100 distribution	teacher top-100 distribution	sampled tokens	teacher top-100 distribution
Training steps	10,000	10,000	up to 4,000	4,000
Checkpoint	step 10,000	step 10,000	lowest conversation KL (App. B.4)	step 4,000

Optimization. Table 4 lists the optimization settings.

B.4 Evaluation and checkpoint selection

Conversation KL. Evaluation uses the full conversation depth: the opening dilemma followed by nine generated user responses. At each token of a user response, we compare the model's next-token distribution when conditioned on the user's prompt P with its distribution under the method being evaluated. All conditions receive the same history and the same response prefix, and differ only in the system prompt. The divergence is the mean of the forward and reverse KL over the full vocabulary. We average it over the tokens of each response, then over the responses of each conversation, and then over all 800 conversations. Every user has the same number of conversations and responses, so this equals the average over users. The token that ends a response is not scored. Confidence intervals are 95% paired bootstrap intervals over conversations (2,000 resamples).

Checkpoint selection. $\mathrm{ALP}_{\mathrm{dist}}$, $\mathrm{ALP}^{\mathrm{off}}_{\mathrm{dist}}$ and $\mathrm{CONV}_{\mathrm{dist}}$ are evaluated at fixed steps (Table 4). For $\mathrm{CONV}_{\mathrm{SFT}}$ we evaluate a series of checkpoints and report the one with the lowest conversation KL. Selection uses the same 16 topics that are reported, so the selected values are optimistic. We do not select on validation loss, because it favours later checkpoints when data is scarce. Table 5 compares the two rules over 36 training runs. With 25 conversations, selecting on validation loss would raise the reported KL by up to 0.010; for example, with two turns per conversation the KL minimum is 0.2120 at step 600, while validation loss picks step 800, which scores 0.2182. From 50 conversations upward the cost is about 0.005 at most.

Checkpoint selection across topics. We test whether a checkpoint selected on some topics also performs well on others. For each run, we select the checkpoint with the lowest conversation KL on $M \in \{1, 2, 4, 8\}$ of the 16 evaluation topics and evaluate it on the remaining topics. We measure the increase in KL relative to the best checkpoint on those remaining topics. We repeat this for every possible selection of M topics across all configurations in Table 3, using the same checkpoint budgets as the main comparison. The mean increase is below 0.00023, and the 95th percentile falls from 0.00111 with one selection topic to 0.00045 with eight (Figure 8a). Both remain below the median range across optimization seeds. Although the selected checkpoint differs from the best checkpoint on the remaining topics in 36% of cases with one selection topic and 15% with eight, these changes have little effect on conversation KL (Figure 8b).

B.5 Compute

All jobs run on a single NVIDIA H200 GPU (141 GB) in bf16. The experiments reported in this paper used about 550 GPU-hours. Training $\mathrm{ALP}_{\mathrm{dist}}$ for 10,000 steps on all 50 users takes about 14 GPU-hours, one $\mathrm{CONV}_{\mathrm{SFT}}$ run takes 2.5–11.5 GPU-hours, and evaluating one run's series of checkpoints takes 1.4–3 GPU-hours.

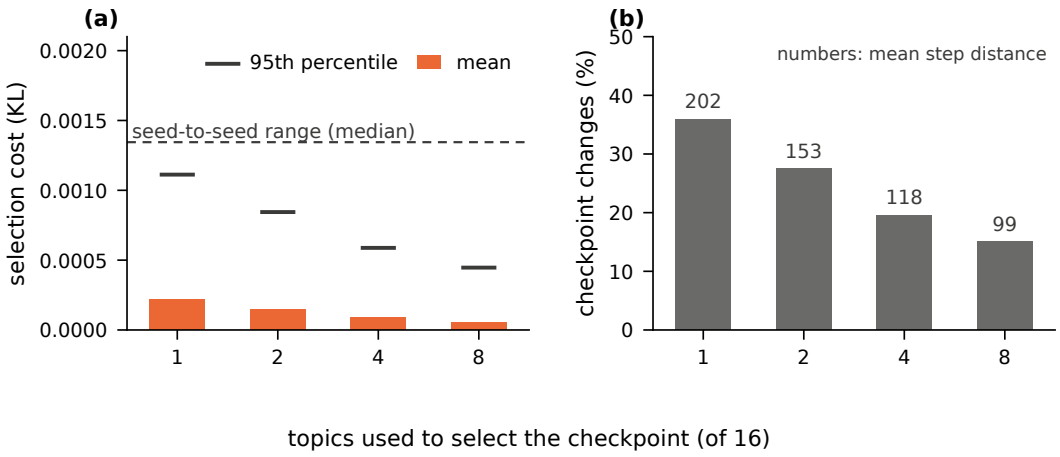


Figure 8: Checkpoint selection across topics, over all breadth–depth configurations and all choices of M selection topics. **(a)** Increase in conversation KL on the remaining topics relative to their best checkpoint. Bars show the mean, ticks the 95th percentile, and the dashed line the median range across three optimization seeds for four configurations. **(b)** Share of cases in which the selected checkpoint differs from the best checkpoint on the remaining topics. Numbers give the mean distance in steps, including cases with no change.

Table 5: Cost of selecting $\mathrm{CONV}_{\mathrm{SFT}}$ on validation loss instead of conversation KL: conversation KL at the validation-loss minimum minus the conversation-KL minimum, over the runs at each number of conversations.

Conversations	Runs	Median cost	Max cost
25	7	0.0062	0.0097
50	11	0.0007	0.0051
100	8	0.0012	0.0041
200	10	0.0007	0.0040

C TEXT-BASED ALTERNATIVES

Shared conversational evidence. We compare the text-based alternatives with observational soft prompts using the same selected conversations for each user, at budgets of $N \in \{1, 5, 10, 25\}$. Smaller budgets are nested within larger ones, using selection seed 1337. All methods observe the same three generated user responses per conversation, together with the opening dilemma and preceding partner replies, corresponding to nominal four-turn depth.

Inferred prompt ($\hat{P}$). We instruct the frozen Qwen3.5-9B model to infer a second-person system prompt from the observed conversations, without access to the original prompt P. The instruction requests both personal characteristics, including background, values and preferences, and writing style. Conversations are rendered as labelled transcripts, with the supplied opening dilemma explicitly distinguished from messages written by the user. Generation uses greedy decoding with thinking disabled.

In-context learning (ICL). For ICL, we place the same labelled conversation transcripts directly in the system prompt, preceded by the instruction:

> You are the person labelled User in the conversations below. Behave as them.

Below {n_label} between one specific person and an assistant.

The person's messages are the ones marked "User:". Across every conversation they were written by the SAME single person - one consistent individual with a specific background, life situation, values, personality and way of writing. The "Assistant:" messages are a generic assistant and tell you nothing about that person except what they were responding to.

{observations}

Your task: read the User messages closely and write the system prompt that would make an assistant role-play as exactly that person.

Requirements for the prompt you write:

- Address the person in the second person, starting with "You are ...".
- Cover WHO they are: their background, life circumstances, what they care about, their values, priorities, opinions and preferences - everything you can infer about the person behind the messages.
- Cover HOW they write: register and tone, sentence length and structure, formatting habits, level of detail, recurring turns of phrase or quirks.
- Be specific and committal. Infer from the evidence rather than hedging; do not describe the assistant, the topics, or this task.
- Write it as a single block of prose, in the imperative/descriptive voice a system prompt uses.

Output ONLY the system prompt text, with no preamble, title, quotes or commentary.

Figure 9: Instruction used to infer $\hat{P}$. The placeholder {n_label} is replaced by "is one conversation" or "are N conversations"; {observations} contains the labelled transcripts.

Table 6: OLMo self-play: conversation KL and questionnaire agreement (%) with OLMo conditioned on P.

Method	Conversation KL
Default assistant	0.754
Cross-persona reference	0.277
Cross-style reference	0.397
$\mathrm{ALP}_{\mathrm{dist}}$	0.233
$\mathrm{CONV}_{\mathrm{SFT}}$	0.265
$\mathrm{CONV}_{\mathrm{dist}}$	0.181

D OLMo Model Family Ablation

We repeat the recovery experiment with OLMo-3-7B-Instruct as the synthetic user, conversation partner, and learner. We regenerate the training and evaluation data with OLMo while keeping the rest of the setting fixed. The conversational methods use nominally 200 conversations per user at four-turn depth.

The soft-prompt method ordering in conversation KL matches the Qwen experiment: $\mathrm{CONV}_{\mathrm{SFT}}$ falls below both cross-user references, $\mathrm{ALP}_{\mathrm{dist}}$ achieves lower KL than $\mathrm{CONV}_{\mathrm{SFT}}$, and $\mathrm{CONV}_{\mathrm{dist}}$ performs best (Table 6).

E Conversation-Partner Ablation

We test whether observational recovery depends on the model used as the conversation partner. We replace the Qwen3.5-9B conversation partner with Qwen3.5-27B or OLMo-3-7B-Instruct, keeping the synthetic user and learner fixed as Qwen3.5-9B. The partner prompt and sampling settings remain unchanged. Each model uses its own chat template.

For OLMo, we regenerate training and evaluation conversations using the same dilemmas and user prompts as in the main experiments. We train five-token $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$ profiles with nominally 200 conversations per user at four-turn depth, using the same training recipe and a 4,000-

Table 7: Conversation KL on evaluation conversations generated with Qwen3.5-9B or OLMo-3-7B-Instruct as partner. Conversational profiles use nominally 200 training conversations at four-turn depth. $\mathrm{CONV}_{\mathrm{SFT}}$ selects step 4000 for the Qwen-trained profile and step 3800 for the OLMo-trained profile on both evaluation sets; $\mathrm{CONV}_{\mathrm{dist}}$ uses step 4000 and $\mathrm{ALP}_{\mathrm{dist}}$ step 10,000. A dash denotes an unscored condition.

		Evaluation partner	
Method	Training partner	Qwen3.5-9B	OLMo-3-7B
Default assistant	–	0.498	0.533
Cross-persona reference	–	0.283	0.331
Cross-style reference	–	0.286	0.295
$\mathrm{ALP}_{\mathrm{dist}}$	–	0.112	0.127
$\mathrm{CONV}_{\mathrm{SFT}}$	Qwen3.5-9B	0.161	0.177
$\mathrm{CONV}_{\mathrm{SFT}}$	OLMo-3-7B	0.158	0.175
$\mathrm{CONV}_{\mathrm{dist}}$	OLMo-3-7B	0.091	0.104

step budget. The leak filter removes 10 of the 10,000 generated training conversations. We evaluate both the Qwen-partner and OLMo-partner $\mathrm{CONV}_{\mathrm{SFT}}$ soft prompts on both evaluation sets, selecting their checkpoints separately on each set. All methods within an evaluation set are scored on the same user responses.

Changing the training partner produces small differences in conversation KL on both evaluation sets (Table 7). Both $\mathrm{CONV}_{\mathrm{SFT}}$ profiles remain below the cross-persona and cross-style references on each set. Changing the evaluation partner shifts the scores more than changing the training partner: both profiles, the default assistant, and $\mathrm{ALP}_{\mathrm{dist}}$ have higher KL on OLMo-partner conversations. On this set, $\mathrm{CONV}_{\mathrm{dist}}$ retains its advantage over $\mathrm{ALP}_{\mathrm{dist}}$.

For Qwen3.5-27B, we compare nominal conversation budgets of 25, 50 and 100 at two- and five-turn depth, evaluating both training conditions on conversations generated with either partner. Checkpoints are selected from the steps evaluated in both conditions. Differences from the Qwen3.5-9B training partner remain below 0.002 nats in every comparison, with no consistent advantage for either partner (Table 8). These results use a single training seed per condition for both partner ablations.

Table 8: Conversation KL for profiles learned with either conversation partner, evaluated on conversations generated with a 9B or 27B partner.

		9B evaluation partner		27B evaluation partner	
N	Depth	Train: 9B	Train: 27B	Train: 9B	Train: 27B
25	2	0.21198	0.21207	0.21610	0.21595
50	2	0.19522	0.19562	0.19942	0.20039
100	2	0.17927	0.18055	0.18328	0.18461
25	5	0.18555	0.18406	0.18936	0.18787
50	5	0.17525	0.17347	0.17847	0.17706
100	5	0.16791	0.16961	0.17111	0.17263

F Prompt Geometry

For each learned soft prompt, we subtract the constant initialization and average over the token positions:

$$\bar{\Delta} = \frac{1}{k}\sum_{j=1}^{k}(z_j - z_{\mathrm{init}}). \tag{3}$$

We measure grouping by style and persona using cosine-distance silhouette scores computed in the full mean-displacement space.

We analyze the five-token $\hat{z}$ profiles trained on 200 conversations at four-turn depth, using checkpoints every 100 steps through step 4000.

Style question: Is the delivery getting here before six?

A: Delivery occurs before six, because couriers begin such routes early in the afternoon and finish well ahead of any evening cut-off.
B: I'm sure it's here before six - I've had nothing but good luck with that courier, and honestly I'm happy about it.
C: before 6 probably? i think they're usually early. might slip a bit idk
D: I hear that van on the street around five - you can see it from the wind0w. It lands before six.
E: Your van will reach the street before six. The driver will ring your bell, and you will hear it from your kitchen.

Unrelated question: Pick a mug for your desk:

A: a small espresso-size one
B: a big one you refill less
C: a plain white one
D: one with a lid

Figure 10: Examples of hand-written questionnaire items in their original option order. The three-option version of the unrelated question omits "one with a lid".

Projection axes. At the final checkpoint, we select the two styles whose centroids are furthest apart, and likewise select two personas. We construct the style axis from the difference between style centroids, excluding the selected personas, and the persona axis from the difference between persona centroids, excluding the selected styles. We remove the component of the persona axis along the style axis and normalize both directions. These axes remain fixed when projecting earlier checkpoints.

G Questionnaire Evaluation

We evaluate whether the learned user models reproduce the synthetic user's answers. The questionnaire combines opinion-survey items from SimBench Pop (Hu et al., 2026) with hand-written style-choice and unrelated questions. Style questions offer alternative replies to the same question, written in different styles; unrelated questions concern choices without any intended connection to the persona or style descriptions. For each question, we select the answer label with the highest next-token probability among the available options and score agreement with P. Figure 10 illustrates the hand-written style and unrelated questions.

Question selection. We select 350 questions from SimBench Pop, taking two disjoint draws of 175 questions, each balanced across Afrobarometer, ESS, ISSP, LatinoBarometro and OpinionQA. The hand-written bank contains 50 style questions and 50 unrelated questions, with the latter evaluated in both three- and four-option versions.

We retain questions whose answers under P vary with persona or style. For each question, we represent each persona by its most frequent answer across styles, and each style by its most frequent answer across personas. A question is retained if at least 20% of these representative answers differ from the most common answer along either axis.

Scoring and uncertainty. We evaluate each question in two option orders and average agreement with P across both orders and all 50 users. We report 95% percentile confidence intervals from 2000 bootstrap resamples of questions, keeping both option orders and the three- and four-option versions of each question together.

H Learning from Modified Conversational Observations

In the main experiments, the learner observes the user's messages in their original form. Here we ask whether changing those messages makes the user harder to model. We call these changes *masking*: inserting extra text, replacing selected tokens, or deleting them. We train new $\mathrm{CONV}_{\mathrm{SFT}}$

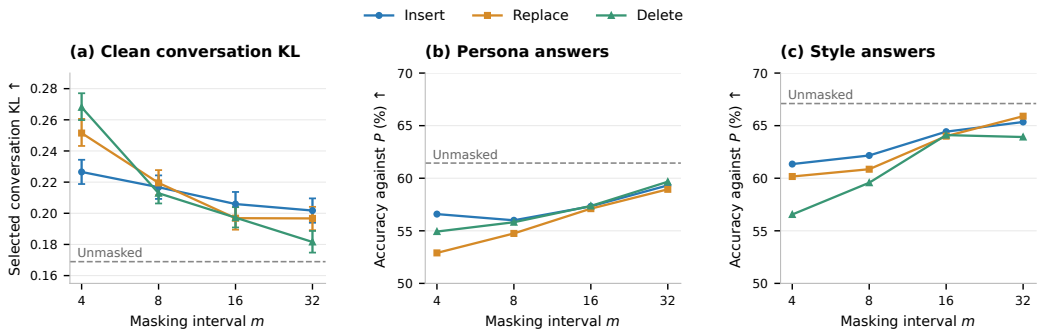


Figure 11: Modified training observations weaken recovery on clean inputs. **(a)** Best evaluated conversation KL; error bars are pointwise 95% topic-bootstrap intervals from 1000 resamples, conditional on selected checkpoints and one training seed. **(b–c)** Persona and style answer accuracy against P at step 3000, on the fixed question subset selected from the unmasked profile's answers; point estimates only. Dashed lines show the unmasked control. Larger m means less frequent modification.

profiles on the modified conversations, using the same training procedure, and then evaluate them on unmodified conversations and questionnaires.

Training with masked observations. Each training run uses exactly one operation of insertion, replacement, or deletion and one fixed masking interval $m \in \{4, 8, 16, 32\}$, giving twelve separate masking conditions. The operations are not combined. Insertion adds the marker § after every mth original tokenizer token; replacement substitutes `[REDACTED]` for that token; deletion removes it. Positions reset within each message. Each operation is applied to generated user turns both as prediction targets and in subsequent training histories; the opening dilemma and assistant replies remain unchanged. For each of the 50 users and each condition, we train a new five-token $\mathrm{CONV}_{\mathrm{SFT}}$ on nominally 100 conversations at five-turn depth. All conditions use the response-prediction objective in Eq. 1 and share the optimization settings, seed, and 3000-step budget with an unmasked control.

Clean evaluation. After training, we evaluate the masked-training $\mathrm{CONV}_{\mathrm{SFT}}$ in place of the standard profile, using the same conversation-KL and questionnaire scoring procedures on unmodified inputs. All masked-training profiles and the unmasked control receive identical clean conversations, questions, and answer keys. Conversation KL uses each condition's best of ten evaluated checkpoints. For questionnaires, all profiles use step 3000. We apply the 20% variability rule from Appendix G to the unmasked control's order-0 answers, rather than P's, retaining 168 persona and 49 style questions. This baseline-selected subset is held fixed across all masking conditions; accuracy remains agreement with clean P answers, pooling both option orders.

Results. All twelve conditions worsen best-checkpoint KL and lower endpoint persona and style agreement (Figure 11). Even insertion impairs recovery despite retaining the original content, so degradation cannot be attributed solely to removing content. Substantial agreement nevertheless remains: persona accuracy is approximately 61.4% after unmodified training and 52.9% under replacement at $m = 4$, still above the default assistant's 41.3%.

I Generated-Conversation Attribution and Disclosure

Generated conversations. To let a user model generate conversations, we repeat the procedure used for the evaluation conversations (Appendix B.4), placing the learned soft prompt or text baseline in the synthetic user's system prompt instead of P. Every method produces 800 conversations (one per user and held-out dilemma), each with 10 user turns.

Judge. A judge, Qwen3.6-27B, reads only the user's side of each conversation: the opening dilemma and the nine user responses. We use an LLM rather than human annotators because the comparison spans 5,600 conversations, which must follow identical criteria for every method; the judge is a different and larger model than the generator, so that it does not recognize its own outputs. As with our synthetic users (Section 7), this favors control over realism: it measures recognizability to an LLM reader, not necessarily to a person. The judge independently classifies the user's persona and style from a list of candidate texts that make up P, labelled by letter in a fixed order. It decodes greedily with thinking disabled and gives its answer first, followed by a justification of at most two sentences. Chance is 10% for persona, 20% for style and 2% for both.

Table 9: Attribution accuracy (%) of generated conversations, 800 per method, with 95% bootstrap intervals over topics (1,000 resamples). Joint requires both answers to be correct.

Method	Persona	Style	Joint
P	96.9 [95.2, 98.2]	84.0 [82.5, 85.5]	80.9 [78.5, 83.1]
$\hat{P}$	95.9 [94.5, 97.1]	50.4 [48.6, 51.7]	47.6 [45.1, 49.8]
ICL	90.8 [88.1, 93.4]	54.5 [51.1, 58.4]	49.4 [46.4, 52.6]
$\mathrm{CONV}_{\mathrm{SFT}}$	91.0 [88.1, 93.4]	80.0 [78.9, 81.1]	72.0 [69.2, 74.5]
$\mathrm{CONV}_{\mathrm{dist}}$	93.8 [92.4, 95.1]	82.0 [80.9, 83.2]	76.0 [74.1, 77.8]
$\mathrm{ALP}_{\mathrm{dist}}$	38.0 [35.0, 41.6]	58.1 [55.5, 60.9]	22.5 [20.2, 25.0]
$\mathrm{ALP}_{\mathrm{dist}}^{\mathrm{off}}$	51.8 [47.4, 56.1]	77.1 [75.9, 78.5]	38.8 [35.5, 42.2]

Attribution accuracy. Conversations generated with $\mathrm{CONV}_{\mathrm{SFT}}$ and $\mathrm{CONV}_{\mathrm{dist}}$ are attributed to the correct persona and style almost as often as those generated under P (Table 9). With ICL and $\hat{P}$, persona attribution remains comparably high, but style attribution falls to about half of the conversations: observed conversations and inferred descriptions convey who the user is more reliably than how the user writes. $\mathrm{ALP}_{\mathrm{dist}}$ has the lowest persona attribution accuracy at 38.0%, despite being closer to P in conversation KL than $\mathrm{CONV}_{\mathrm{SFT}}$ (0.1122 against 0.1607). $\mathrm{ALP}_{\mathrm{dist}}^{\mathrm{off}}$ improves persona and style attribution relative to $\mathrm{ALP}_{\mathrm{dist}}$, but its persona accuracy remains far below that of every method with access to the user's conversations.

What persona attribution relies on. The judge's justifications indicate that persona attribution rests mainly on biographical content. For conversations generated under P, the judge typically cites details that users disclose about themselves, for instance: "The user explicitly mentions working as a sales representative, wanting money for expensive sneakers, and enjoying hobbies like baking chocolate chip cookies ...". Under $\mathrm{ALP}_{\mathrm{dist}}$, such details are rarer, and the judge more often infers the persona from the manner of writing: "The user's informal, conversational tone ... and focus on immediate financial stress ... align with Candidate B's profile as a young adult", in this case incorrectly. A lexical count supports this reading. The proportion of user responses containing a word found only in the user's own persona description ranks the methods in the same order as persona attribution, and the ranking is robust to the details of the count. The count registers only explicit mentions, yet it agrees with the justifications: $\mathrm{ALP}_{\mathrm{dist}}$'s conversations are harder to attribute largely because they reveal less about the user.

J Behavioral-Change Detection: Evaluation Details

Configuration. We reuse the five-token $\mathrm{CONV}_{\mathrm{SFT}}$ profiles trained on nominally 200 conversations at four-turn depth at step-4000 from Figure 2(c). For each user and change type, one eligible replacement prompt is sampled (seed 2026) and kept across all topics and both switch positions. Generation reuses parameters specified in section B.1.

Data and calibration. The 16 held-out topics are split deterministically by topic identifier into seven calibration and nine test topics, shared by all 50 users. At each switch position, each change-type comparison contains 450 changed and 450 unchanged responses; the same unchanged responses serve all three comparisons. For each scorer, user, and position, we normalize losses by subtracting the mean and dividing by the sample standard deviation of the seven unchanged calibration responses. For each scorer and position, the detection threshold is the empirical 95th percentile

of its 350 normalized calibration scores, using linear interpolation; scores above it are classified as changed. Neither normalization nor threshold selection uses changed or test responses.

Scope. At a target calibration false-positive rate of 5%, the trained profile's test false-positive rates are 8.0% before response 1 and 11.3% before response 3. Strong ranking therefore does not guarantee reliable threshold transfer.