TOWARD VALID AGENT EVALUATION: RECONCILING TASK FIDELITY AND HUMAN PLAUSIBILITY IN LLM-BASED INTERACTIVE USER SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM-based user simulators enable scalable evaluation of interactive agents, but task success is informative only when the simulated interaction itself provides valid evidence of agent capability. This can fail in two ways: the simulator may drift from the assigned task or it may exhibit behavior that is extreme relative to real users. These risks motivate our joint assessment of Task Fidelity (TF) and Human Plausibility (HP). We audit 30 user LLMs, finding that 34.1% of successful interactions contain HP violations and 45.9% of failures contain TF violations. An intuitive response is to prompt simulators with personas, which we find improves HP but often at the expense of TF. To address the risks, we propose a method that separates what the simulator needs to accomplish from how it pursues those goals. Across 30 user LLMs, this separation turns average TF loss into a gain ($-6.4 \rightarrow +0.6$ pp), while further improving HP ($+6.1$ pp). These gains extend to agent evaluation, with reduced score sensitivity to user LLM choice, more consistent rankings, and strong agreement with external evaluations. Together, these findings show that task-faithful, human-plausible interactions provide a stronger basis for interpreting agent outcomes as evidence of capability.

1 INTRODUCTION

LLM-based user simulators enable scalable and reproducible evaluation of interactive agents, whose tasks require coordinating with users over multiple turns(Sekulić et al., 2024; Niu et al., 2024; Dou et al., 2025). Unlike tasks that interact only with a static problem or environment—such as math, QA, or code generation—these tasks depend on user behavior, which shapes the conditions under which the agent is evaluated. These evaluations typically summarize performance using task success, yet an observed success or failure is informative about agent capability only if the interaction producing it provides a valid test of the intended capability.

Two distinct failures in simulated interactions can undermine this interpretation. A simulator assigned to request an exchange may instead accept a refund, producing a natural dialogue that no longer tests the intended task (Mehri et al., 2026b). Conversely, a task-faithful simulator may make requests so explicit that the agent succeeds without resolving the ambiguity common in human interaction (Zhou et al., 2026b). These failure modes motivate two validity conditions for simulated interactions: ***Task Fidelity***, whether the interaction remains faithful to the assigned scenario, and ***Human Plausibility***, whether the simulator's behavior remains within a plausible range of human interaction.

We operationalize these conditions as rollout-level checks and use them to audit interactions generated by 30 user LLMs. The audit reveals substantial validity risks hidden by agent task outcomes alone: 34.1% of successful interactions violate the Human Plausibility check, while 45.9% of failed interactions violate the Task Fidelity check. The former finding weakens the interpretation of success as evidence of performance with real users and the latter makes failure difficult to attribute to the agent. Moreover, violation rates vary substantially across user LLMs, suggesting that measured agent performance may reflect in part simulator limitations rather than agent capability.

Improving simulator-based evaluation therefore requires reducing both types of validity failures rather than addressing either in isolation. A natural approach is to prompt user-simulating LLMs

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.

with personas derived from human behavior. We find that this approach substantially improves Human Plausibility, **but often at the cost of Task Fidelity** (-6.4 pp on average), reducing one validity risk while increasing another. The challenge is therefore ensuring realistic behavior without departing from the assigned task.

We therefore explore whether this trade-off can be reduced by separating *what the user needs to accomplish* from *how it pursues those goals*. Our framework first tracks what remains to be accomplished under the assigned goals, then generates the user's next utterance guided by a persona derived from human conversations. This separation is designed to let the persona shape how the interaction unfolds without changing the assigned task. This separation mitigates the trade-off: averaged over 30 user LLMs, TF recovers the loss from persona prompting ($-6.4 \rightarrow +0.6$ pp), while HP improves further ($+6.1$ pp).

Finally, we examine whether these improvements strengthen the conclusions drawn about agents. With the factorized approach, **scores more strongly reflect differences among agents** (33–47% $\rightarrow$ about 90% of score variation), while rankings are more stable across user LLMs and remain consistent with two external evaluations of interactive task performance and human preferences. Together, these findings show that user simulator choice and design matter for reliable agent comparisons, highlighting the importance of the underlying interactions' validity in simulator-based evaluation.

2 Criteria for Evaluating Task Outcomes in User–Agent Interactions

An LLM-based user simulator plays two roles at once in interactive task benchmarks. It **instantiates** the task on which an agent is evaluated, while also **standing in for the person** with whom the agent is expected to interact. To assess task fidelity and Human Plausibility (HP), we check whether an interaction preserves the assigned scenario and stays within human-reference behavioral bounds. We use binary verdicts because interpreting a task outcome requires distinguishing whether task or behavioral violations occurred, rather than assigning a degree of fidelity or plausibility.

We consider an interaction between a user simulator $\mathcal{U}_s$ conditioned on an assigned scenario $s = (q, c)$ and an agent $\mathcal{A}$. The task specification q states what the user seeks to accomplish and how acceptable resolutions depend on the course of the dialogue. The user context c contains task-relevant private information and constraints available to the user. Their multi-turn interaction generates the rollout $\tau_{\mathcal{U}_s,\mathcal{A}} = (u_1, a_1, \ldots, u_T, a_T)$, where u_t and a_t denote the simulator and agent turns at round t. We write τ when $\mathcal{U}_s$ and $\mathcal{A}$ are clear from context.

2.1 Task Fidelity

Task fidelity (TF) assesses whether the simulator preserves the correspondence between the benchmark-assigned task and the task the agent encounters as the dialogue develops. We allow variation in wording and how requests are conveyed across turns, so that expressing the same task content differently is not treated as a TF violation. Additionally, since TF evaluates the user's adherence to the assigned scenario, not the agent's task performance, we instruct the judge to treat agent errors alone as insufficient evidence of a TF violation. We define task fidelity as the binary, LLM-judged rollout-level verdict $\mathrm{TF}(s, \tau) \in \{0, 1\}$, with

$$\mathrm{TF}(s, \tau) = 1 \quad \Longleftrightarrow \quad \text{the user's behavior in } \tau \text{ is task-consistent with } s.$$

- **Pass** ($\mathrm{TF} = 1$)**.** The user's behavior remains faithful to the goals, information, and constraints specified in the scenario.
- **Violation** ($\mathrm{TF} = 0$)**.** The user fails to convey required request, agrees to outcome inconsistent with the scenario, violates the scenario's information requirements, or ends the dialogue before task resolution.

We compare combinations of verdicts from three LLM judges on 500 human-labeled dialogues and select the combination with the highest agreement with human TF labels (Appendix D.1).

2.2 HUMAN PLAUSIBILITY

While task fidelity is judged against the assigned scenario, human plausibility has no analogous specification: how people behave in these interactions is available only as observed behavior. We therefore assess **Human plausibility (HP)** using 13,633 multi-turn human–AI dialogues from PRISM (Kirk et al., 2024), ThoughtTrace (Jin et al., 2026), and ConvApparel (Meshi et al., 2026) as a reference for human interaction behavior.

We operationalize this comparison as a calibrated, episode-level boundary test in the following two steps:

1. Indicator selection. The 19 indicators introduced by Zhou et al. (2026b) describe user behavior—for example, *front-loading* (concentrating information in early turns) and *clarification questions* (asking for clarification). Among these, we prioritize interaction patterns that influence how the dialogue unfolds, excluding surface-style measures. We further restrict the selection to indicators supporting stable cross-corpus calibration. Each retained indicator j summarizes a rollout's user behavior as a numerical value $z_j(\tau)$, computed from word counts or fixed lexical patterns rather than LLM judgment. For example, front-loading measures the share of user word tokens in the first two turns (for dialogues with at least three user turns).

2. Boundary calibration. For each retained indicator we calibrate an upper threshold only, since human values on these indicators sit on the lower side (Appendix D.2). We choose this threshold so that at most $1-\gamma$ of human dialogues are incorrectly flagged as violations, using $\gamma \in \{0.95, 0.99\}$.[1] We denote the resulting threshold for indicator j as $\theta_\gamma^{(j)}$. For example, under the $\gamma = 0.95$ setting, front-loading is flagged when more than 92.7% of the user's total words appear in the first two turns. Refer to Appendix D.2 for indicator selection and threshold calibration.

We define the binary rollout-level verdict as

$$\mathrm{HP}_\gamma(\tau) = 1 \quad \Longleftrightarrow \quad z_j(\tau) \le \theta_\gamma^{(j)} \text{ for every applicable retained axis } j.$$

- **Pass** ($\mathrm{HP}_\gamma = 1$)**.** The rollout remains within the calibrated human boundary on every applicable retained axis.
- **Violation** ($\mathrm{HP}_\gamma = 0$)**.** The rollout exceeds the calibrated human boundary on at least one applicable retained axis.

This is a one-sided upper-bound check, not a similarity score. A pass thus means only that no measured behavioral indicator is unusually high relative to the reference corpus, not that the simulator reproduces the full distribution of human behavior. We assess the boundary through **human false-positive rates** and sensitivity to the reference. With indicators fixed, person-level five-fold recalibration yields a mean false-positive rate of **3.32%**, while the boundary flags **2.83%** of 495 held-out in-domain human dialogues (Zhou et al., 2026b). Switching to an in-domain reference yields **89.0% agreement** on simulator verdicts (**Cohen's $\kappa = 0.78$**; Appendix D.3).

2.3 JOINT INTERPRETATION

The same task outcome can have different implications for agent performance depending on the interaction's task fidelity and human plausibility (Figure 1). Task-faithful, human-plausible interactions (**green**) provide clearer evidence of agent capability, whereas violations of both requirements (**red**) make outcomes unreliable indicators of agent performance. Neither requirement alone, however, is sufficient. Task-faithful but *behaviorally implausible* interactions (**yellow**, lower left) can give **false assurance** about performance with real users. Conversely, human-plausible but *task-unfaithful* interactions (**yellow**, upper right) create a **task mismatch** that complicates attributing the outcome to the agent.

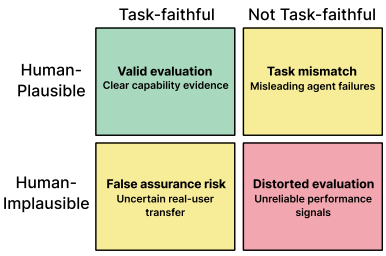


Figure 1: Joint implications of the two requirements for evaluation validity.

[1] We use $\gamma = 0.95$; see Appendix D.2 for justification.

In this way, the joint verdicts make the two validity requirements empirically examinable. We next ask how often these risks accompany benchmark successes and failures, using the joint verdicts to audit the interactions underlying these outcomes.

3 User Simulator–Induced Risks to Evaluation Validity

Reward alone can conceal task mismatch or give false assurance about an agent's performance with real users. Here, we audit dialogues between user simulators and agents to examine how often agent successes and failures occur alongside TF or HP violations by the user simulator, and how these rates vary with the choice of LLM used as the user simulator.

3.1 Experimental Setup

We study all 114 tasks from the retail domain of τ^2-bench (Barres et al., 2025), where each rollout pairs a user simulator $\mathcal{U}_s$ with a tool-using customer-service agent $\mathcal{A}$. Each user LLM is prompted with the assigned scenario $s = (q, c)$, without an additional persona specification. We determine agent success using the benchmark's binary reward, which checks whether the agent achieves the specified task outcome. We analyze successes and failures using the joint TF and HP verdicts defined in Section 2.

3.2 Interpreting Agent Successes and Failures

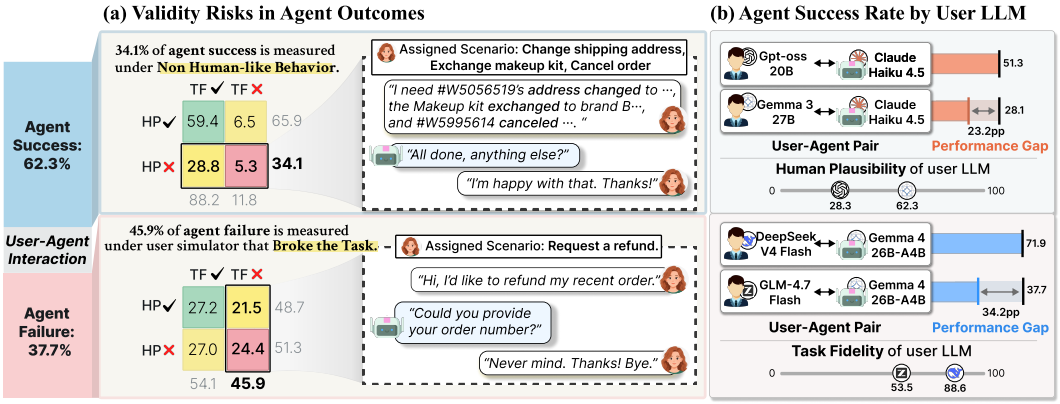


Figure 2: **TF–HP verdicts for agent successes and failures.** (a) Percentages within successes or failures across 30 user LLMs, with three trials per task and DeepSeek-V4-Flash as agent. (b) Agent success rates and user-LLM HP/TF for the agent–user pairs shown.

Figure 2 shows the joint TF and HP verdicts separately for agent successes and failures. Both groups include rollouts in which the user simulator violates these requirements, raising the following concerns for interpreting agent performance:

1. Success can pose a false assurance risk. 34.1% of agent successes co-occur with HP violations in the user simulator's behavior. For example, the agent may succeed with a simulator that gives an unusually long opening message or repeats the same information excessively. Such success does not by itself establish comparable performance with real users.

2. Failure can be difficult to attribute. Conversely, in **45.9%** of agent failures, the user simulator violates TF by deviating from the assigned scenario. For example, the simulator may end the conversation while required requests remain unresolved. These failures cannot be attributed solely to the agent without further examination. This does not imply that task drift caused every such failure.

3. Agent scores vary with the user LLM choice. Even with the agent and tasks held fixed, changing the user LLM can substantially alter the agent's success rate. The two examples in Figure 2(b) show success-rate gaps of 23.2 and 34.2 percentage points, accompanied by contrasting HP and TF

pass rates. These differences raise concern that agent scores may reflect the conditions created by the user simulator, rather than the agent's capabilities alone.

3.3 The Task-Fidelity Cost of Persona Prompting

Adding an explicit persona is a straightforward way to guide a user simulator toward more plausible human behavior. We therefore supplement the assigned scenario with a persona derived from human dialogues (Section 4.1), which we refer to as *task+persona prompting* (T+P), and compare it with the scenario-only setup of Section 3.1, which we refer to as *task-only prompting* (TO).

On average across the 30 user LLMs on the 114 retail tasks, T+P brings user behavior within the human-reference bounds more often (HP: 59.4% $\rightarrow$ 84.2%), while simulators deviate from the assigned task more frequently (TF: 75.4% $\rightarrow$ 69.6%, -5.8 pp; -6.4 pp with the airline tasks included, Section 4.3). HP improves for nearly all models, but at a TF cost in 22 of them, including both large-scale and small/mid-scale models. A human evaluation is consistent with this cost: 39.5% of T+P dialogues show both persona expression and a TF violation, versus 11.0% of matched TO dialogues (Appendix E.6).

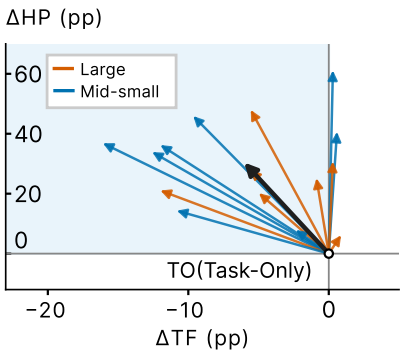


Figure 3: A subset of 15 user LLMs. TO baselines are normalized to $(0, 0)$; black marks the mean across displayed models.

These findings motivate a simulator design that allows persona-driven behavior while remaining faithful to the assigned scenario.

4 Separating Task and Persona in User Simulation

The results above show that persona prompting improves HP, but often at the expense of TF. To address this trade-off, we separate task requirements from persona-driven behavior, keeping what users need to accomplish fixed while leaving room for how they pursue it. In our approach, we first derive persona profiles from human dialogues to capture behavioral and linguistic variation (Section 4.1). We then generate each user turn in two stages: the Goal Tracker updates the current task state and identifies the goals to pursue next, and the Persona Realizer selects persona-conditioned conversational behavior and realizes it as an utterance (Section 4.2).

4.1 Persona Extraction and Representation

Step 1: Extracting Contrastive Observations from Human Dialogues. To ground personas in observed user behavior, we draw on 492 human–agent dialogues from the τ-bench study of Zhou et al. (2026b), in which multiple human users were assigned the same task scenarios and interacted with an agent. To identify behavior that is specific to an individual rather than prescribed by the task, we compare each human dialogue against other dialogues generated under the same scenario. For each human dialogue, we form five same-scenario comparisons. Two come from other human users assigned to the same scenario, allowing us to identify behavioral and linguistic differences among people. The remaining three are LLM-simulated dialogues generated from the same task specification and user context, using GPT-5.6-Terra, DeepSeek-V4-Flash, and Kimi-K2.6. These comparisons help identify human-observed behaviors that may be absent from typical simulator behavior.

For each dialogue pair, an LLM extractor identifies behavioral and linguistic differences. To avoid encoding task requirements as personal traits, we instruct the extractor to exclude behaviors that the human user exhibited as directed by the assigned scenario. Across the five pairs, extraction yields an average of 20.4 distinguishing observations per person (7.5 behavioral and 12.9 linguistic).

Step 2: Representing Individual Users through Persona Profiles. Here, we turn the extracted observations into a persona profile that captures how each person behaves and speaks. An LLM first

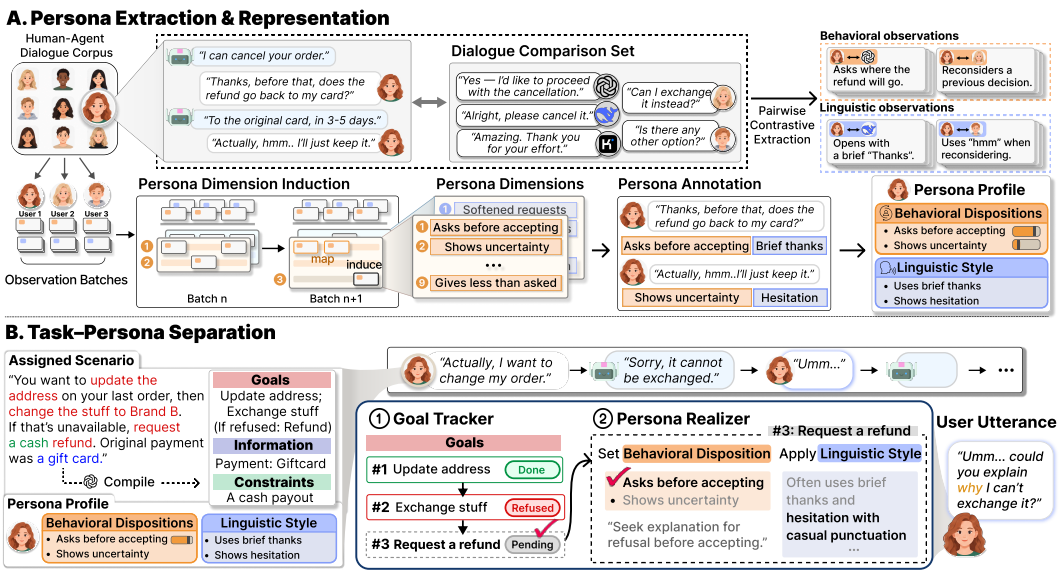


Figure 4: ***Top:*** Persona extraction and representation. ***Bottom:*** Task–Persona Separation with a Goal Tracker and a Persona Realizer.

derives behavioral and linguistic categories from a subset of the extracted observations, then processes the remaining observations using this growing category set, assigning them to existing categories or defining new ones when needed. We then manually select and consolidate these categories into a shared set of *persona dimensions*, comprising nine behavioral dispositions *(e.g., delegation and clarification)* and twelve linguistic styles *(e.g., terse replies and softened requests)*.

We then use an LLM to label each user utterance in the original dialogue with any applicable persona dimensions and aggregate these labels into a persona profile. Behavioral dispositions are marked as *high* if observed more than once, *moderate* if once, and *none* if unobserved. For linguistic style, we retain patterns that occur relatively frequently in the person's user turns compared with other profiles exhibiting those patterns. On average, we identify 3.7 behavioral dispositions and 5.0 linguistic styles per person. Note that each profile summarizes tendencies observed in a single dialogue, rather than stable behavioral probabilities. Refer to Appendix C.1 for definitions of the persona dimensions and the criteria used to construct individual profiles.

4.2 Task–Persona Separation for User Generation

Given the assigned scenario $s = (q, c)$, a persona profile, and the dialogue history $\tau_{<t}$, our simulator $\mathcal{U}_s$ generates the next user utterance u_t in two stages (Figure 4), separating **what the user needs to accomplish** from **how they pursue it**. Before the dialogue begins, we compile the scenario s into a *fixed task structure* $\tilde{s}$ that the Goal Tracker uses to track progress during the interaction. At each turn, **(1) Goal Tracker** updates the *current task state* using the compiled structure $\tilde{s}$ and $\tau_{<t}$, identifying which goals remain to be pursued. **(2) Persona Realizer** determines how the user responds based on the profile's behavioral dispositions, such as answering only part of a request, and generates u_t in the profile's linguistic style.

Task Preparation. We prepare a fixed task structure $\tilde{s}$ for each scenario s, organizing the user's goals and the conditions under which they change. Prepared once from the scenario and shared across user LLMs, this structure provides the basis for tracking task progress during interaction. We use GPT-5.6-Terra for this preparation. Refer to Appendix E.3.1 for the justification of offline compilation and the effect of the compiled structure on TO and T+P.

Step 1: Goal Tracking. The Goal Tracker uses an LLM to interpret the dialogue history $\tau_{<t}$ against $\tilde{s}$, identifying which requests have been resolved so far and which transition conditions have been met. Based on this interpretation, the tracker updates the task state and identifies the goals to pursue next by traversing $\tilde{s}$.

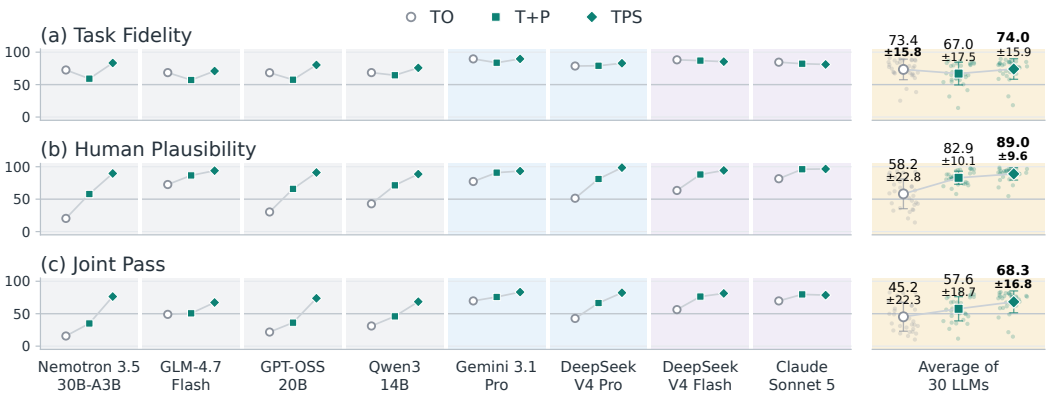


Figure 5: **Task fidelity and human plausibility across user LLMs.** Eight selected models (left); individual model rates and unweighted means ± model SD across 30 LLMs (right).

Step 2: Persona Realization. Given the current task state and the agent's current utterance, the Persona Realizer determines a behavioral response from the list of behavioral dispositions defined for the persona. For example, when the agent asks for information, the behavioral disposition "*Gives less than asked*" may lead the user to provide only part of it. It expresses this response as the next user utterance u_t, grounded in the task state maintained by the Goal Tracker and following the linguistic style captured in the persona profile. Each disposition's inclusion probability is adjusted based on how frequently it appeared in the original dialogue. See Appendix C.2 for details.

Together, the two stages keep the assigned task as the stable basis for generation while allowing the persona to shape how the user pursues it. We next evaluate whether this separation improves TF and HP while preserving persona-specific variation in the resulting interactions.

4.3 Experimental Evaluation

Setup. We compare Task–Persona Separation (TPS, Ours), described in Section 4.2, with TO (task-only prompting) and T+P (task+persona prompting) on the 114 retail and 50 airline tasks of τ^2-bench (Barres et al., 2025). We use the same 30 user LLMs and evaluation protocol as in Section 3, with the same persona profiles for T+P and TPS. For the persona-conditioned methods (T+P and TPS), each task is run once with each of its three persona profiles; for TO, each task is run for three trials. We compute TF, HP, and joint pass rates (passing both TF and HP) for each model. Figure 5 summarizes these rates across all 30 models alongside eight illustrative model-level examples. Refer to Appendix B for model configurations and Appendix E.1 for full results.

Mitigating the TF–HP Trade-off. On average, TPS recovers the TF lost under T+P while further improving HP (Figure 5). Across the user LLMs, TF returns to the TO level (74.0% vs. 73.4%, compared with 67.0% for T+P), while HP increases beyond T+P (82.9% $\rightarrow$ 89.0%). Together, these gains show that task fidelity and human plausibility need not come at each other's expense: TPS increases the share of interactions satisfying both requirements (57.6% $\rightarrow$ 68.3%).

Beyond these average gains, TPS narrows the gap across model scales: gains over T+P are larger for small/mid-scale models than for large-scale ones, reducing the joint-pass gap between the two groups from 25.0 to 15.4 pp (Table 34). This convergence in task fidelity and human plausibility offers a basis for agent evaluation that is less dependent on user LLM choice, which we examine in Section 5.

Persona Identifiability. Higher HP pass rates do not necessarily imply that simulated users retain individual differences: a simulator could satisfy the measured human-plausibility bounds while collapsing distinct personas into a generic behavioral pattern. We therefore examine whether different personas can be distinguished from simulated retail dialogues. Given a simulated user–agent dialogue and the three persona profiles constructed in Section 4.1 from human dialogues on the same task, an LLM judge selects the profile that best matches the simulated user's behavioral dispositions and linguistic style (three-way classification; random chance: 33%).

We provide profiles containing behavioral dispositions alone (Behav.-only) or with linguistic style (Full). TO remains near chance (Table 1), and persona prompting achieves higher identification accuracy but also incurs a loss in TF. TPS retains identifiable personas, with accuracy remaining well above chance even when profiles contain only behavioral dispositions (Appendix E.5). Even the original human dialogues yield only 70.5% (Full) and 57.6% (Behav.-only) identification accuracy, as the three persona profiles for a given task are not always clearly distinguishable.

Table 1: Persona identification accuracy (%) over 30 user LLMs, mean ± task SD.

Method	Full	Behav.-only
TO	33.8 ± 4.6	33.4 ± 5.0
T+P	66.9 ± 12.8	50.8 ± 11.2
TPS	61.1 ± 11.3	46.1 ± 8.8
Human dialogues	70.5 ± 23.8	57.6 ± 25.6

5 Contributions to Agent Evaluation

We now examine whether addressing these risks strengthens the basis for interpreting individual user–agent interaction outcomes and supports more reliable agent comparisons. We evaluate 10 agent LLMs against each of 8 user LLMs on 114 retail tasks, with one rollout per task and method.

5.1 Improving the Validity of Individual Interactions

We examine how simulation can mitigate the interaction-level validity risks from Section 3. Under TPS, more successes and failures arise from interactions passing both TF and HP (Figure 6(a)), strengthening the basis for interpreting either outcome; successes better reflect human-plausible behavior, while clarity in failure attribution lost when eliciting more human-like behavior is restored.

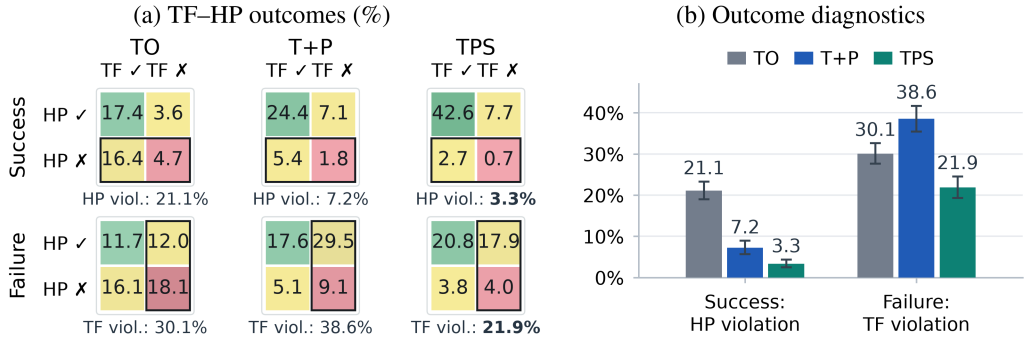


Figure 6: **Mitigating validity risks in agent successes and failures.** (a) Joint TF–HP percentages of all scored rollouts, averaged equally across agent–user conditions. (b) Outlined risks: successes with HP violations and failures with TF violations. Error bars: task-bootstrap 95% CIs.

Figure 6(b) summarizes the implications for interpreting outcomes. Across TO/T+P/TPS, HP-violating successes decrease (21.1% → 7.2% → 3.3% of scored rollouts), while TPS reverses the increase in TF-violating failures, lowering them below TO (30.1% → 38.6% → 21.9%). Appendix F.2 details the aggregation and uncertainty estimates. We next ask whether the agent comparisons drawn from these outcomes are also more robust.

5.2 From Valid Rollout to Valid Evaluation

We next examine whether the agent success rates obtained by aggregating these interaction outcomes support reliable comparisons. We retain the original benchmark outcomes without TF/HP filtering or reweighting, and examine the sources of score variation, ranking robustness, and external agreement.

Score Decomposition. For each simulation method, a descriptive two-way ANOVA decomposes score variation into agent, user-LLM, and residual components. Under direct prompting, user LLM choice contributes a comparable or greater share of score variation than agent LLM choice, whereas under TPS, about 90% of variation is attributable to the agent LLM versus 33–47% under direct prompting (Figure 7(a)). Absolute score sensitivity also decreases: the mean within-agent SD across user LLMs is 14.8 → 11.8 → 6.1 percentage points (Appendix F.1).

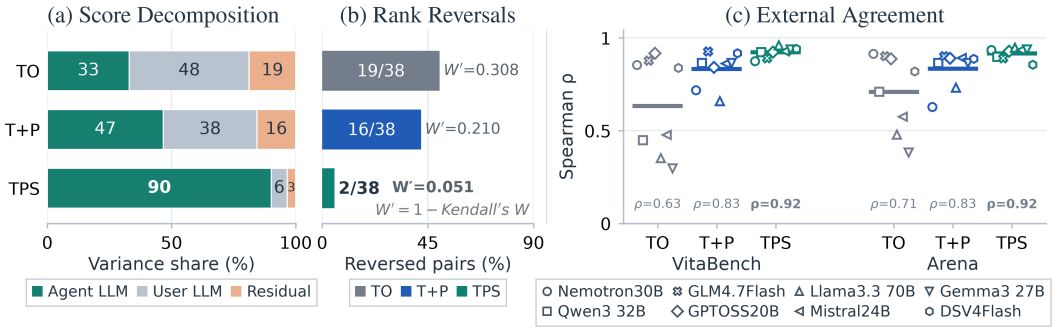


Figure 7: **Simulator dependence and agreement with external evaluations.** (a) Score decomposition. (b) Ranking robustness across user LLMs. (c) External agreement; markers denote user LLMs, while bars and numbers show mean correlations.

Ranking Robustness. For each simulation method, we assess how much agent rankings change across user LLMs using two complementary measures: pairwise ordering reversals and Kendall's W. Under TPS, only 2 of the 38 common agent pairs (mean-score gaps $>$ 5pp) reverse order across user LLMs, compared with 16–19 under direct prompting. Full rankings likewise show stronger agreement under TPS (tie-corrected $W = 0.95$ versus 0.69–0.79; Figure 7(b) displays $W' = 1-W$). Appendix F.3 reports all-pair results and cutoff sensitivity.

External Agreement. Consistency across user LLMs does not itself establish agreement with external evaluations. We therefore compare them with two benchmarks: VitaBench (He et al., 2026), a task-oriented evaluation with an LLM-based user simulator, and Arena (LMArena, 2026), a human-preference evaluation based on real-user pairwise votes. Under TPS, agreement remains high across all eight user LLMs (Spearman $\rho \geq 0.85$ for both references; Figure 7(c)). Although neither reference establishes prediction of real-user task success, TPS maintains strong external agreement without depending on which of the user LLMs is used. See Appendix F.4 for evaluation details.

6 Related Work

LLM-based user simulators LLM user simulators support agent evaluation and training (Ni et al., 2026; Barres et al., 2025; Kong et al., 2024). Research pursues goal consistency through explicit tracking (Dongre et al., 2025; Mehri et al., 2026b) and behavioral realism (Zhou et al., 2026b) through persona conditioning (Shim et al., 2026; Chopra et al., 2026) or training on human conversations (Naous et al., 2026; Wu et al., 2026; Wang et al., 2026). We quantify the TF–HP trade-off, separate task tracking from persona realization, and evaluate the effects on agent scores and rankings.

Human-likeness of user simulators No accepted criterion defines sufficient realism for user simulators (Schatzmann et al., 2006; Pietquin & Hastie, 2013). Existing evaluations assess human–simulator *distinguishability* (Hathidara et al., 2026; Meshi et al., 2026), linguistic and behavioral *similarity* to human data (Ivey et al., 2026; Mehri et al., 2026a), and transfer of simulator-based conclusions to real users (*proxy validity*; Zhang & Balog, 2020; Seshadri et al., 2026). Our verdict is a limited distinguishability test: whether a rollout falls outside the behavioral range observed in human conversations.

7 Conclusion

Interpreting simulated outcomes as evidence of agent capability requires assessing the validity of the underlying interactions. Our joint assessment of task fidelity and human plausibility reveals validity risks hidden by success rates. We find that separating task requirements from persona-driven behavior offers a promising path toward improving the validity of simulator-based agent evaluation. This approach supports agent comparisons that are less sensitive to the choice of user LLM while remaining consistent with external evaluations. Direct validation against real-user task performance remains an important next step.

AI USE STATEMENT

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

ETHICS STATEMENT

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

REPRODUCIBILITY STATEMENT

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

REFERENCES

Hua Ai and Diane Litman. Assessing dialog system user simulation evaluation measures using human judges. In *Proceedings of ACL-08: HLT*, pp. 622–629, 2008.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. τ^2-bench: Evaluating conversational agents in a dual-control environment, 2025. URL https://arxiv.org/abs/2506.07982.

Douglas Biber. *Variation across speech and writing*. Cambridge university press, 1991.

Penelope Brown and Stephen C. Levinson. *Politeness: Some Universals in Language Usage*. Cambridge University Press, 1987.

Harshita Chopra, Kshitish Ghate, Aylin Caliskan, Tadayoshi Kohno, Chirag Shah, and Natasha Jaques. Beyond cooperative simulators: Generating realistic user personas for robust evaluation of llm agents. *arXiv preprint arXiv:2605.12894*, 2026.

Herbert H. Clark and Susan E. Brennan. Grounding in communication. In Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley (eds.), *Perspectives on Socially Shared Cognition*, pp. 127–149. American Psychological Association, 1991.

Vardhan Dongre, Ryan A Rossi, Viet Dac Lai, David Seunghyun Yoon, Dilek Hakkani-Tür, and Trung Bui. Drift no more? context equilibria in multi-turn llm interactions. *arXiv preprint arXiv:2510.07777*, 2025.

Yao Dou, Michel Galley, Baolin Peng, Chris Kedzie, Weixin Cai, Alan Ritter, Chris Quirk, Wei Xu, and Jianfeng Gao. Simulatorarena: Are user simulators reliable proxies for multi-turn evaluation of ai assistants? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 35200–35278, 2025.

Kate Forbes-Riley and Diane Litman. Benefits and challenges of real-time uncertainty detection and adaptation in a spoken dialogue computer tutor. *Speech Communication*, 53(9–10):1115–1136, 2011.

Kallirroi Georgila, James Henderson, and Oliver Lemon. User simulation for spoken dialogue systems: Learning and evaluation. In *Proc. Interspeech 2006*, pp. paper–2035, 2006.

Howard Giles and Tania Ogay. Communication accommodation theory. In Bryan B. Whaley and Wendy Samter (eds.), *Explaining Communication: Contemporary Theories and Exemplars*, pp. 293–310. Lawrence Erlbaum, 2007.

H. Paul Grice. Logic and conversation. In Peter Cole and Jerry L. Morgan (eds.), *Syntax and Semantics, Vol. 3: Speech Acts*, pp. 41–58. Academic Press, 1975.

Ashutosh Hathidara, Julien Yu, Vaishali Senthil, Sebastian Schreiber, and Anil Babu Ankisettipalli. Mirrorbench: A benchmark to evaluate conversational user-proxy agents for human-likeness. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 8978–8988, 2026.

Wei He, Yueqing Sun, Hongyan Hao, Xueyuan Hao, Zhikang Xia, Qi Gu, Hui Su, and Xunliang Cai. Vitabench: Benchmarking llm agents with versatile interactive tasks in real-world applications. In *International Conference on Learning Representations*, volume 2026, pp. 91654–91681, 2026.

Jonathan Ivey, Shivani Kumar, Jiayu Liu, Hua Shen, Sushrita Rakshit, Rohan Raju, Haotian Zhang, Aparna Ananthasubramaniam, Junghwan Kim, Bowen Yi, et al. Real or robotic? assessing whether llms accurately simulate qualities of human responses in human-llm dialogue. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 41395–41432, 2026.

Chuanyang Jin, Binze Li, Haopeng Xie, Cathy Mengying Fang, Tianjian Li, Shayne Longpre, Hongxiang Gu, Maximillian Chen, and Tianmin Shu. Thoughttrace: Understanding user thoughts in real-world llm interactions. *arXiv preprint arXiv:2605.20087*, 2026.

Hannah R Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, et al. The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *Advances in Neural Information Processing Systems*, 37:105236–105344, 2024.

Chuyi Kong, Yaxin Fan, Xiang Wan, Feng Jiang, and Benyou Wang. Platolm: Teaching llms in multi-round dialogue via a user simulator. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7841–7863, 2024.

LMArena. Arena Leaderboard Dataset. Hugging Face dataset, 2026. URL `https://huggingface.co/datasets/lmarena-ai/leaderboard-dataset`. Text style-control, overall category; leaderboard snapshot published 13 September 2026.

Shuhaib Mehri, Philippe Laban, Sumuk Shashidhar, Marwa Abdulhai, Sergey Levine, Michel Galley, and Dilek Hakkani-Tür. Measuring and mitigating the distributional gap between real and simulated user behaviors. *arXiv preprint arXiv:2605.07847*, 2026a.

Shuhaib Mehri, Xiaocheng Yang, Takyoung Kim, Gokhan Tur, Shikib Mehri, and Dilek Hakkani-Tür. Goal alignment in llm-based user simulators for conversational ai. *Transactions of the Association for Computational Linguistics*, 14:852–876, 2026b.

Ofer Meshi, Krisztian Balog, Sally Goldman, Avi Caciularu, Guy Tennenholtz, Jihwan Jeong, Amir Globerson, and Craig Boutilier. Convapparel: A benchmark dataset and validation framework for user simulators in conversational recommenders. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5270–5304, 2026.

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.

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.

Björn Nieth, Marianna Gracheva, Michaela Mahlberg, Bjoern Eskofier, and Emmanuelle Salin. How human-like are large language models? a register-aware linguistic evaluation framework. *arXiv preprint arXiv:2605.23651*, 2026.

Cheng Niu, Xingguang Wang, Xuxin Cheng, Juntong Song, and Tong Zhang. Enhancing dialogue state tracking models through llm-backed user-agents simulation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8724–8741, 2024.

Nicolò Pagan, Petter Törnberg, Christopher A Bail, Anikó Hannák, and Christopher Barrie. Computational turing test reveals systematic differences between human and ai language. *arXiv preprint arXiv:2511.04195*, 2025.

Olivier Pietquin and Helen Hastie. A survey on metrics for the evaluation of user simulations. *The knowledge engineering review*, 28(1):59–73, 2013.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828, 2021.

Yada Pruksachatkun, Elaine Wan, Lyanna Chen, Kai-Wei Chang, and Chien-Sheng Wu. Salessim: Benchmarking and aligning multimodal language models as retail user simulators. *arXiv preprint arXiv:2605.08334*, 2026.

Alex Reinhart, Ben Markey, Michael Laudenbach, Kachatad Pantusen, Ronald Yurko, Gordon Weinberg, and David West Brown. Do llms write like humans? variation in grammatical and rhetorical styles. *Proceedings of the National Academy of Sciences*, 122(8):e2422455122, 2025.

Jost Schatzmann, Karl Weilhammer, Matt Stuttle, and Steve Young. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The knowledge engineering review*, 21(2):97–126, 2006.

Ivan Sekulić, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, Andre Ferreira Manso, and Roland Mathis. Reliable llm-based user simulator for task-oriented dialogue systems. In *Proceedings of the 1st Workshop on Simulating Conversational Intelligence in Chat (SCI-CHAT 2024)*, pp. 19–35, 2024.

Preethi Seshadri, Samuel Cahyawijaya, Ayomide Odumakinde, Sameer Singh, and Seraphina Goldfarb-Tarrant. Lost in simulation: Llm-simulated users are unreliable proxies for human users in agentic evaluations. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 47423–47439, 2026.

Jeonghoon Shim, Woojung Song, Cheyon Jin, Seungwon KooK, and Yohan Jo. Non-collaborative user simulators for tool agents. In *International Conference on Learning Representations*, volume 2026, pp. 64486–64534, 2026.

Gabriel Skantze. Exploring human error recovery strategies: Implications for spoken dialogue systems. *Speech Communication*, 45(3):325–341, 2005.

Joseph Suh, Ayush Raj, Minwoo Kang, and Serina Chang. Quantifying the utility of user simulators for building collaborative llm assistants. *arXiv preprint arXiv:2605.09808*, 2026.

Kuang Wang, Xianfei Li, Shenghao Yang, Li Zhou, Feng Jiang, and Haizhou Li. Know you first and be you better: Modeling human-like user simulators via implicit profiles. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21082–21107, 2025.

Yingshan Susan Wang, Cedegao E Zhang, Linlu Qiu, Zexue He, Pengyuan Li, Alex Pentland, Roger P Levy, and Yoon Kim. Learning user simulators with turing rewards. *arXiv preprint arXiv:2606.19336*, 2026.

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

Se-eun Yoon, Zhankui He, Jessica Echterhoff, and Julian McAuley. Evaluating large language models as generative user simulators for conversational recommendation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1490–1504, 2024.

Shuo Zhang and Krisztian Balog. Evaluating conversational recommender systems via user simulation. In *Proceedings of the 26th acm sigkdd international conference on knowledge discovery & data mining*, pp. 1512–1520, 2020.

Xuhui Zhou, Weiwei Sun, Weihua Du, Jiarui Liu, Haojia Sun, Qianou Ma, Tongshuang Wu, Yiming Yang, and Maarten Sap. Odyssim: Building foundation models for human behavior simulation. *arXiv preprint arXiv:2606.14199*, 2026a.

Xuhui Zhou, Weiwei Sun, Qianou Ma, Yiqing Xie, Jiarui Liu, Weihua Du, Sean Welleck, Yiming Yang, Graham Neubig, Sherry Tongshuang Wu, et al. Mind the sim2real gap in user simulation for agentic tasks. *arXiv preprint arXiv:2603.11245*, 2026b.

A LIMITATION

Our formulation of Human Plausibility assesses selected behavioral patterns rather than the full distribution of human behavior. HP is designed for a narrower purpose: identifying interactions whose measured behavior departs sufficiently from observed human behavior that the resulting agent outcome warrants caution. Its binary verdict records whether such a departure occurs in a rollout, rather than assigning a continuous *human-likeness* score.

Within this scope, we evaluate HP against human dialogue data and prior human-likeness measures, and examine its sensitivity to the choice of human reference. These evaluations provide empirical support for the boundary's alignment with observed human behavior and its use as an interpretable rollout-level diagnostic of evaluation validity risks. Direct validation against real-user task performance remains an important next step.

B EXPERIMENTAL SETUP

Table 2 lists the 30 user LLMs of the simulator comparison, and Table 3 the models of the agent evaluation. User LLMs run with their provider's default sampling settings, kept the same across TO, T+P, and TPS; where a checkpoint exposes a reasoning toggle, reasoning is on. Every episode is capped at 60 orchestrator steps of τ^2-bench, where a step is one message passed between the user, the agent, and the environment; an episode that reaches the cap is scored as is.

Table 2: The 30 user LLMs of the simulator comparison, with the Hugging Face checkpoint for locally run models. The 12 API models form the *large-scale* group: proprietary models, whose parameter counts are undisclosed, and open-weight models of several hundred billion parameters (DeepSeek, Kimi, Mistral Large 3). The 18 locally run open-weight models (4–70B parameters) form the *small/mid-scale* group.

Large-scale (12)

- GPT-5.6 Terra: API
- GPT-5.6 Sol: API
- Claude Sonnet 5: API
- Claude Haiku 4.5: API
- Gemini 3.1 Pro: API
- Gemini 3.5 Flash Lite: API
- DeepSeek V4 Pro: API
- DeepSeek V4 Flash: API
- DeepSeek V3.2: API
- Kimi K2.6: API
- Kimi K2.5: API
- Mistral Large 3: API

Small/mid-scale (18)

- Llama 3.3 70B: `meta-llama/Llama-3.3-70B-Instruct`
- Qwen3 32B: `Qwen/Qwen3-32B`
- OLMo 3.1 32B: `allenai/Olmo-3.1-32B-Instruct`
- Gemma 4 31B: `google/gemma-4-31B-it`
- Nemotron 3.5 30B-A3B: `nvidia/NVIDIA-Nemotron-3.5-Lightning-30B-A3B-BF16`
- GLM-4.7 Flash: `zai-org/GLM-4.7-Flash`
- Qwen3.8 27B: `Qwen/Qwen3.8-27B`
- Gemma 3 27B: `unsloth/gemma-3-27b-it`
- Gemma 4 26B-A4B: `google/gemma-4-26B-A4B-it`
- Mistral Small 24B: `mistralai/Mistral-Small-24B-Instruct-2501`
- GPT-OSS 20B: `openai/gpt-oss-20b`
- Qwen3 14B: `Qwen/Qwen3-14B`
- Qwen3.5 9B: `Qwen/Qwen3.5-9B`
- Qwen3 8B: `Qwen/Qwen3-8B`
- Llama 3.1 8B: `meta-llama/Llama-3.1-8B-Instruct`
- Ministral 8B: `mistralai/Ministral-8B-Instruct-2410`
- Qwen3.5 4B: `Qwen/Qwen3.5-4B`
- Qwen3 4B: `Qwen/Qwen3-4B`

Table 3: The ten agent and eight user LLMs of the agent evaluation.

Agent LLMs	Claude Sonnet 5, Gemini 3.1 Pro, Kimi K2.6, DeepSeek V3.2, Claude Haiku 4.5, Gemma 4 26B-A4B, GLM-4.7 Flash, Mistral Small 24B, Nemotron 3.5 30B-A3B, Llama 3.3 70B
User LLMs	Nemotron 3.5 30B-A3B, Qwen3 32B, GLM-4.7 Flash, GPT-OSS 20B, Llama 3.3 70B, Mistral Small 24B, Gemma 3 27B, DeepSeek V4 Flash

Model selection for the agent evaluation. The user LLMs are drawn from the pool of Table 2. Seven are open-weight models between 20B and 70B parameters, one from each of seven model families, so that the evaluation does not hinge on one family or one size; the eighth, DeepSeek V4 Flash, is an API model and adds a stronger user. The ten agent LLMs span the range from large-scale API models to open-weight models of the same mid-size range.

C Method Details

C.1 Details on Persona Extraction and Representation

This section follows the two steps of Section 4.1: extracting distinguishing observations from human dialogues, and turning them into persona profiles. We use GPT-5.6-Sol for contrastive extraction, category induction, and turn-level annotation.

Step 1: Contrastive extraction. Each of the 492 human dialogues is paired with five others on the same scenario, three with LLM user simulators and two with other human participants, and an LLM extractor lists how the person's behavior and wording differ from the partner's. Table 4 gives the resulting counts; the per-person averages in Section 4.1 are these totals divided by the number of compared dialogues. Because each pair is extracted separately, a difference that holds against several partners is recorded several times.

Extraction quantity	Count
Human dialogues	492
Dialogue pairs	2,460
Behavioral observations	3,699
Linguistic observations	6,332
Persona profiles	492

Table 4: Extraction coverage. Each compared dialogue contributes five pairs; observation counts sum over pairs.

The extractor is instructed to skip responses the scenario itself prescribes, such as changing the request when the agent asks for confirmation, and to report only choices the person made on their own. Observations that change a goal or constraint, contradict the user's facts, or accept an outcome the scenario does not allow are likewise excluded, since they are task violations rather than dispositions.

Step 2: Persona dimensions and profiles. An LLM induces candidate behavioral and linguistic categories from the extracted observations. We manually select and consolidate these candidates into the 21 persona dimensions in Table 5.

For behavior, we reduce 16 categories to nine by merging overlapping categories and excluding behaviors judged common in task-only simulators. For linguistic style, we consolidate similar categories, reducing the list from 24 to 12.

A separate pass annotates each person's original dialogue with these dimensions turn by turn, again setting aside scenario-scripted actions. The profile is built from these turn annotations rather than from the pairwise observations, so its extents reflect the dialogue itself.

Observed extent. For a disposition, we count the user turns in which it appears, at most once per turn: zero, one, and two or more turns give *none*, *moderate*, and *high*. For a linguistic style, we divide its annotated turns by the person's total user turns; a share below the median among profiles showing that style is *moderate*, and at or above it *high*. The style description in the profile lists the linguistic styles marked high; absence is stated only for greetings, brief thanks, and softened requests, where its absence is itself informative. Appendix H.8 walks through one dialogue from

Table 5: The closed set of 21 persona dimensions: nine behavioral dispositions and twelve linguistic styles. Each persona profile records which dimensions were observed and to what extent.

Persona dimension	Description
Behavioral dispositions (9)	
Gives less than asked	Supplies part of the requested information, leaving the rest for later.
Answers loosely	Gives vague, off-format, or off-topic replies instead of the requested answer.
Leaves it to the agent	Leaves record lookup and search scope or order to the agent.
Holds back on requirements	Questions or holds back on details and procedural requirements before complying.
Doesn't take no	Does not accept a refusal or limitation straight away.
Asks before accepting	Seeks explanation, confirmation, or progress before deciding or treating an issue as settled.
Shows uncertainty	Expresses uncertainty or acknowledges a slip, apologizing or asking for help.
Straight to business	Proceeds directly to the task without greetings or transitions.
Brings feelings in	Introduces personal circumstances, feelings, or impatience.
Linguistic styles (12)	
Terse replies	Uses a bare word, value, identifier, or fragment instead of a sentence.
Unpolished writing	Uses typos, lowercase, missing punctuation, or colloquial shorthand.
Blunt imperatives	Gives short commands without softening.
Softened requests	Uses courteous questions or softeners such as "please".
Hesitation	Uses fillers, hedges, or resigned wording when choosing or asking.
Greeting	Opens with a greeting before stating the request.
Brief thanks	Thanks the agent briefly at the end or before moving on.
Effusive thanks	Expresses warm, emphatic, or emotional appreciation.
Voiced frustration	Complains or voices irritation, impatience, or dissatisfaction.
Emotive punctuation	Uses emoticons, stretched spelling, capitals, or repeated punctuation.
Apology	Apologizes before correcting or clarifying.
Playful humor	Uses jokes, playful asides, or self-deprecation.

observations to profile. Table 6 shows six resulting profiles, three people on the same retail task and three on the same airline task, to illustrate how much people assigned the same scenario differ.

Table 6: Persona profiles of three different people on one retail task (top) and on one airline task (bottom), derived from the real human dialogue dataset of Zhou et al. (2026b). Behavioral dispositions are listed with their observed extent; the style description lists the linguistic styles marked high.

Retail task 2, person 1	**Retail task 2, person 2**	**Retail task 2, person 3**
Behavioral dispositions	***Behavioral dispositions***	***Behavioral dispositions***
• Gives less than asked (high)	• Leaves it to the agent (high)	• Gives less than asked (moderate)
• Straight to business (high)	• Gives less than asked (moderate)	• Answers loosely (moderate)
• Holds back on requirements (moderate)	• Answers loosely (moderate)	• Leaves it to the agent (moderate)
• Asks before accepting (moderate)	• Doesn't take no (moderate)	• Holds back on requirements (moderate)
• Shows uncertainty (moderate)	• Shows uncertainty (moderate)	• Asks before accepting (moderate)
Linguistic style	• Straight to business (moderate)	• Straight to business (moderate)
• Terse replies	• Brings feelings in (moderate)	***Linguistic style***
• Blunt imperatives	***Linguistic style***	• Terse replies
	• Terse replies	• Greeting
	• Unpolished writing	• Brief thanks
	• Blunt imperatives	• Emotive punctuation
	• Voiced frustration	
Airline task 32, person 1	**Airline task 32, person 2**	**Airline task 32, person 3**
Behavioral dispositions	***Behavioral dispositions***	***Behavioral dispositions***
• Asks before accepting (high)	• Doesn't take no (high)	• Gives less than asked (high)
• Shows uncertainty (moderate)	• Asks before accepting (high)	• Answers loosely (high)
• Brings feelings in (moderate)	• Gives less than asked (moderate)	• Doesn't take no (moderate)
Linguistic style	• Leaves it to the agent (moderate)	• Shows uncertainty (moderate)
• Softened requests	• Brings feelings in (moderate)	• Brings feelings in (moderate)
• Greeting	***Linguistic style***	***Linguistic style***
	• Unpolished writing	• Greeting
	• Softened requests	• Apology
	• Greeting	
	• Brief thanks	

C.2 Details on Task–Persona Separation for User Generation

This section gives the operating details of the task preparation and the two stages in Section 4.2: what the compiled task structure contains, how the Goal Tracker updates the task state, and what the Persona Realizer receives and returns. Prompt templates and a recorded turn are in Appendix H.

Scenario compilation. Before the dialogue, an LLM reads the scenario and writes a fixed structure $\tilde{s}$: an ordered list of *items*, the separate things the customer came for, each holding a small tree of *wants*. The root want is what the customer seeks at the outset; every other want names the want it replaces and the agent event that triggers the replacement, such as "the agent says the item is out of stock", or is marked as a step the customer takes on their own after voicing the parent. An item can also wait on another item being settled. The compiler copies the scenario's wording and values, never resolves a branch, and never adds an identifier, price, or agent line the scenario does not contain. Instructions about how to talk, such as what to mention first or what not to reveal, are kept beside the structure as talk rules rather than inside it.

The structure is stored once and is not rewritten during the dialogue. In all experiments, GPT-5.6-Terra compiles each scenario once, and every user LLM receives the same structure; Appendix E.3.1 compares this with each user LLM compiling its own scenario.

Step 1: Goal tracking. The Goal Tracker uses an LLM to interpret the dialogue history against the fixed task structure. After each agent turn, the LLM assesses which requests have been expressed or resolved and whether the conditions for a scenario-specified goal change have been met. The tracker updates the current task state from these judgments according to the predefined transition rules, identifying the goals that remain to be pursued. Once no requests remain, the simulator closes the dialogue.

Step 2: Persona realization. The Persona Realizer uses an LLM to select a behavioral response and generate the next user utterance in the persona's linguistic style in a single call. It selects from the persona's dispositions available for the turn, using the current task state and dialogue to determine which apply. The prompt allows at most one disposition, except for *going straight to business* and *bringing feelings in*, which can accompany the chosen response without replacing it.

Each disposition is independently sampled for inclusion in the prompt; inclusion does not require its use. Except for the two accompanying dispositions above, inclusion probabilities decrease after each use reported by the LLM according to the profile's observed extent: *high* follows $1 \to 0.66 \to 0.33 \to 0$, and *moderate* follows $1 \to 0$. Dispositions marked *none* are excluded. Inclusion without use does not reduce the probability. *Going straight to business* and *bringing feelings in* remain available throughout when present in the profile.

D Task Fidelity and Human Plausibility

D.1 Details on Task-Fidelity Judgment

The TF judge reads the assigned scenario and the completed user–agent dialogue, excluding internal tool interactions. It reconstructs the user's requests, confirmations, and factual claims, then checks them against the scenario under the branch realized by the dialogue. Four checks assess whether the user conveyed the required requests, accepted only compatible outcomes, respected the scenario's known and explicitly unknown information, and avoided premature departure. A rollout passes TF when all four checks pass. The judge evaluates the user's task fidelity, not conversational style or the agent's success; an agent error alone does not constitute a user violation. Appendix H.6 provides the full prompt and detailed decision rules.

Selecting the judge combination. We collect human ground-truth TF labels for 500 dialogues and apply three LLM judges—GPT-5.6-Terra, DeepSeek-V4-Flash, and Kimi-K2.6—to the same dialogues. We combine their binary TF verdicts by logical OR (pass $= 1$): the combined verdict is pass if at least one included judge passes, and violation only if all included judges fail. We compare candidate combinations by their agreement with the human labels, measured as the percentage of matching binary TF verdicts. The combination with the highest agreement is selected as our TF judge. Table 7 provides the comparison with the individual judges. These agreement scores are measured on the 500-dialogue selection set, rather than an independent test set.

Judge(s)	Verdict combination rule	Agreement (%)
G	Single judge	80.0
D	Single judge	75.6
K	Single judge	73.2
G + D	$G \vee D$ (Any pass)	93.2
G + K	$G \vee K$ (Any pass)	88.6
D + K	$D \vee K$ (Any pass)	87.6
G + D + K	$G \vee D \vee K$ (Any pass)	91.2

Table 7: TF agreement with human ground truth on 500 dialogues. G: GPT-5.6-Terra; D: DeepSeek-V4-Flash; K: Kimi-K2.6.

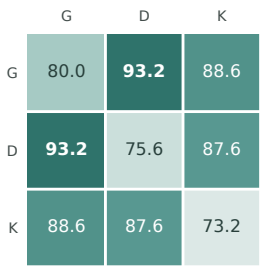


Figure 8: TF agreement (%): single judges on the diagonal; OR pairs off-diagonal.

D.2 Details on Human-Plausibility Indicators and Calibration

Reference corpora. HP is calibrated on human behavior observed outside the benchmark: the three human–AI dialogue corpora in Table 8, none of which involves the benchmark's tasks or agents. Indicator selection and threshold calibration use this reference alone, never simulator pass rates, so the boundary cannot be tuned toward any simulator.

Table 8: Human–AI reference corpora used for HP calibration.

Source	Dialogues	Human turns
PRISM (Kirk et al., 2024)	7,944	26,967
ThoughtTrace (Jin et al., 2026)	2,155	8,529
ConvApparel (Meshi et al., 2026)	3,534	11,764
Total	13,633	47,260

Indicator selection. Table 37 in Appendix I lists the 19 deterministic behavioral statistics we considered, following Zhou et al. (2026b), and how each is treated. Two screens apply. First, the verdict concerns dialogue management rather than surface wording, so the nine axes that measure politeness, formality, emotion, and similar surface style are excluded; words per turn is kept because its upper tail captures reading out the task brief instead of disclosing information interactively. Second, an axis must have a reachable upper-tail boundary and be stable across corpora: information seeking is excluded because its boundary reaches the statistic's maximum, and identifier density fails the cross-corpus screen at 95%, which allows at most 3% of a held-out corpus to be flagged on a single axis. Pushback is withheld for a different reason: the free-dialogue reference contains few policy conflicts, so its zero threshold reflects limited occasions for pushback rather than a human norm against it. Clarification questions and strategy pivots keep nonzero boundaries and are retained.

Boundary calibration. Each indicator $z_j(\tau)$ is computed from the user turns by word counts and fixed lexical matchers, without an LLM, and only unusually high values count as violations. For K retained indicators and confidence level γ, each indicator receives an equal share b of the error budget. With n_j applicable reference dialogues, the threshold $\theta_\gamma^{(j)}$ is the (k_j+1)-th largest reference value, where

$$b = \frac{1-\gamma}{K}, \qquad k_j = \lfloor b\,(n_j+1) \rfloor - 1.$$

A rollout violates an indicator only under strict exceedance, and passes HP when it stays within the boundary on every applicable retained axis:

$$\mathrm{HP}_\gamma(\tau) = \prod_{j \in \mathcal{J}(\tau)} \mathbb{I}\big[\, z_j(\tau) \le \theta_\gamma^{(j)} \,\big],$$

where $\mathcal{J}(\tau)$ is the set of retained indicators applicable to τ. If a human dialogue is exchangeable with the reference, each indicator's exceedance probability is bounded, and the union bound over the K indicators bounds the chance that the dialogue is flagged:

$$\Pr\big[\, z_j(\tau) > \theta_\gamma^{(j)} \,\big] \le \frac{k_j+1}{n_j+1} \le b, \qquad \Pr\big[\, \mathrm{HP}_\gamma(\tau) = 0 \,\big] \le Kb = 1-\gamma.$$

Seven indicators are retained at either level (Table 9): at 99%, repetition reaches its support boundary and drops out, while identifier density passes the cross-corpus screen and enters. We use $\gamma = 0.95$ throughout: at 95% the boundary flags only 2.83% of held-out in-domain human dialogues (Table 10), and raters labeled 87.2% of the dialogues flagged at 95% but not at 99% non-human (Appendix E.7).

Table 9: HP boundaries at the two confidence levels. Verdicts use full precision and strict exceedance; a dash marks an excluded indicator. †Requires at least three user turns.

Indicator	$\gamma = 95\%$	$\gamma = 99\%$
Identity confusion	0.000	0.250
Repetition	0.545	—
Words per turn	58.000	101.000
Opening length	65.000	106.000
Identifier density	—	1.000
Front-loading†	0.927	0.966
Clarification questions	0.333	0.500
Strategy pivots	0.333	0.500

Why an upper boundary only. Figure 9 shows the human reference and the TO rollouts of Section 3 on the eight retained indicators. Six indicators admit no lower boundary: 96–99.7% of reference dialogues have zero repetition, identity confusion, identifier mentions, clarification questions, or strategy pivots, and 5.5% open with a single word, so at the budget of Section 2 a lower threshold falls on that mass and nothing can lie below it. Words per turn and front-loading admit one, at 2.25 words and 0.105 against upper thresholds of 58 words and 0.927; the lower thresholds would flag 0.0% and 0.8% of all simulated rollouts, the upper ones 18.0% and 2.1%. We therefore calibrate upper thresholds only, on every indicator.

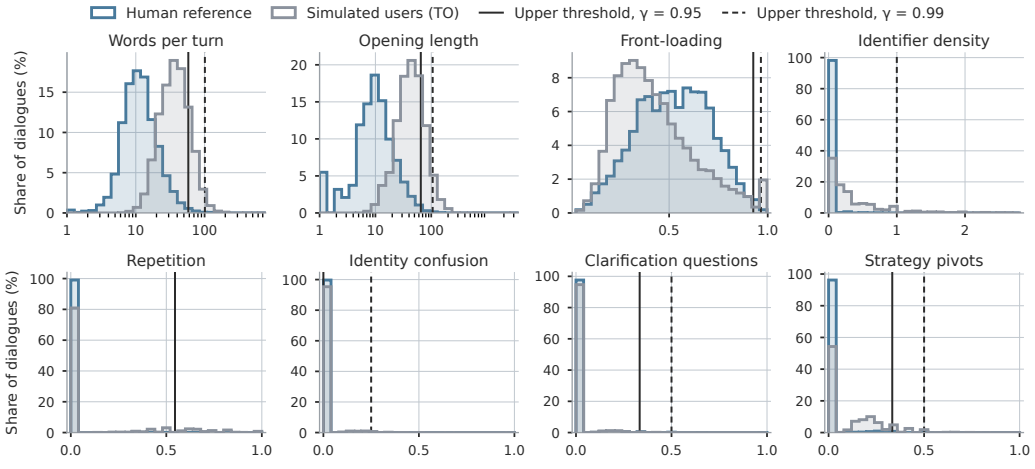


Figure 9: Human reference (blue) and TO rollouts of Section 3 (gray) on the eight indicators retained across the two calibration levels, as shares of dialogues; front-loading uses dialogues with at least three user turns. Vertical lines are the upper thresholds at the two confidence levels.

Which indicators fire. Figure 10 decomposes the HP violations in the TO retail audit of Section 3, the same rollouts used for the reference-sensitivity analysis. A rollout may violate several indicators, so the bars are overlapping prevalences rather than a partition.

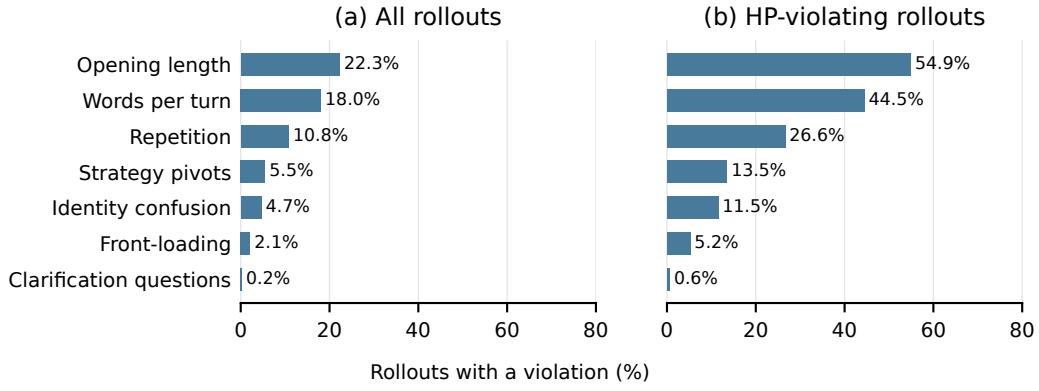


Figure 10: HP violations by indicator among (a) all TO rollouts of the Section 3 audit and (b) the HP-violating ones among them, under the 95% boundary. Categories may overlap.

D.3 Statistical Validation of the HP Boundary

D.3.1 False Positives on Human Dialogues

The boundary is meant to flag behavior people do not show, so its first test is how often it flags people. We apply it to the 495 in-domain human dialogues (Zhou et al., 2026b),[2] which were never used for calibration. At 95%, 14 of them are flagged (2.83%), within the analytic bound of 4.98% (Table 10). The implied human pass rate, 97.17%, is the rate at which people clear the boundary and is a reference point rather than a ceiling on simulators.

Table 10: Share of human dialogues flagged by the HP boundary used throughout the paper. For the reference corpus, used for calibration, the value is the analytic bound for a new dialogue exchangeable with it; for the in-domain human dialogues, never used for calibration, it is observed.

Human dialogues	Sources	n	$\gamma = 95\%$	$\gamma = 99\%$
Reference corpus	PRISM, ThoughtTrace, ConvApparel	13,633	$\leq$4.98%	$\leq$0.97%
In-domain human dialogues	τ^2-bench retail and airline	495	2.83%	0.20%

Stability of the calibration procedure. We also recalibrate the boundary on part of the reference, with the retained indicators fixed, and apply it to the held-out part: five person-level folds (Table 11) and each corpus in turn (Table 12). All held-out rates stay within the budget except ThoughtTrace. Without it, the reference is smaller and lacks its long user turns, so the words-per-turn boundary drops from 58 to 50 words; the boundary used in the paper includes all three corpora.

Table 11: Person-level five-fold: recalibrated on four folds, applied to the fifth.

Held-out fold	n	$\gamma = 95\%$	$\gamma = 99\%$
Fold 1	2,756	3.19%	0.44%
Fold 2	2,727	4.07%	0.66%
Fold 3	2,717	3.61%	0.88%
Fold 4	2,691	2.97%	0.67%
Fold 5	2,742	2.77%	0.69%
Mean		3.32%	0.67%

Table 12: Leave-one-corpus-out: recalibrated on two reference corpora, applied to the third.

Calibrated on	Held out	n	$\gamma = 95\%$	$\gamma = 99\%$
ThoughtTrace + ConvApparel	PRISM	7,944	3.41%	0.78%
PRISM + ConvApparel	ThoughtTrace	2,155	6.36%	2.18%
PRISM + ThoughtTrace	ConvApparel	3,534	1.67%	0.11%

D.3.2 Sensitivity to the Human Reference

A boundary calibrated on different human dialogues would flag different rollouts. We therefore compare the HP boundary, calibrated on the reference corpus, with an *in-domain-calibrated* boundary, calibrated on the 342 retail in-domain human dialogues. Its smaller sample sets each threshold at the calibration maximum, and it retains ten indicators, including identifier density, pushback, and information seeking (Table 38 in Appendix I). We apply both to the TO retail rollouts of the 30 user LLMs in Section 3.

Table 13 shows that the picture in Section 3 survives the change of reference: among agent successes, HP violations occur in 34.1% of rollouts under the HP boundary and 39.1% under the in-domain-calibrated one, and the two boundaries agree on 89.0% of all rollouts (Cohen's $\kappa = 0.78$).

[2]The corpus covers 115 retail tasks, one more than τ^2-bench; the three dialogues on that task are not used for persona extraction, leaving 492.

Table 13: HP verdicts on the TO retail rollouts of the Section 3 audit under the reference-calibrated boundary used in the paper and the in-domain-calibrated boundary, by benchmark outcome. Agreement compares the two boundaries' verdicts on the same rollouts.

		HP violation		Agreement	
	n	Reference-calibrated	In-domain-calibrated	%	κ
Agent Success	6,394	34.1%	39.1%	89.2%	0.77
Agent Failure	3,864	51.3%	57.4%	88.5%	0.77
All	10,258	40.6%	46.0%	89.0%	0.78

D.4 Agreement with Human-Likeness Measures from Prior Work

HP is a rollout-level boundary check: it flags an interaction whose behavior falls outside the human reference, because interpreting an agent's outcome requires knowing whether such behavior occurred (Section 2). It is not designed to score how human-like a rollout is. Prior work scores human-likeness directly, with classifiers, detectors, and distances between text distributions (Pruksachatkun et al., 2026; Reinhart et al., 2025; Nieth et al., 2026; Pillutla et al., 2021; Wang et al., 2025; Pagan et al., 2025). We apply these measures to the same rollouts to check that HP nonetheless points in the same direction as human-likeness.

Setup. We use the eight user LLMs of the agent evaluation (Table 3) and their retail rollouts under TO, T+P, and TPS, run as in Section 4.3, 342 per user LLM and method. We call one user LLM under one method a *simulator*, which gives 24 simulators. The human reference is the 342 in-domain retail dialogues, collected on the same tasks. Every measure reads customer turns only, because the human dialogues were collected with a different agent. Before applying the measures, we remove τ^2-bench control tokens such as `###STOP###`, which appear only in simulated dialogues. The classifiers we train to score single rollouts learn from human dialogues and from TO rollouts of other user LLMs, with the tasks split into two folds, so none of them scores a rollout or a task it was trained on. Table 39 in Appendix I describes each measure and how we apply it.

Reference values and agreement statistics. Each row reports the value for real customers: the classifiers we train score held-out human dialogues, measures that compare two sets of dialogues compare the two halves of the human set, and the remaining measures use all 342 human dialogues. For measures that score single rollouts, a simulator's value is the median over its rollouts. We compare each measure with HP in two ways. ρ is the Spearman correlation across the 24 simulators between the HP pass rate and the measure, signed so that higher means more human-like. AUC is the probability that a rollout HP passes scores as more human-like than a rollout HP flags, over the rollouts of all 24 simulators; it is defined only for measures that score single rollouts.

Table 14: Human-likeness measures from prior work on the 24 simulators. ↑/↓: higher/lower is more human-like under the measure. Bold: human value. Underline: the most human-like of the three methods under the measure. HP pass rates are shown for reference. Method columns average the eight user LLMs.

			Measure value				Agreement with HP	
Measure	Source	Level	TO	T+P	TPS	Human	ρ	AUC
HP pass rate (%)	this paper	rollout	47.1	79.2	91.2	**98.2**	–	–
Features disjoint from HP								
n-gram classifier ↑	Pruksachatkun et al.	rollout	0.27	0.53	0.49	**0.67**	0.60	0.75
Biber classifier ↑	Reinhart et al.	rollout	0.17	0.56	0.56	**0.71**	0.59	0.74
Biber MMD^2 ↓	Nieth et al.	simulator	0.059	0.055	0.033	**0.005**	0.53	–
MAUVE ↑	Pillutla et al.	simulator	0.02	0.11	0.08	**0.95**	0.35	–
USP detector ↑	Wang et al.	rollout	0.29	0.74	0.75	**0.90**	0.75	0.73
CTT accuracy ↓	Pagan et al.	simulator	0.97	0.91	0.88	**0.51**	0.65	–
Features shared with HP								
Sim2Real Dice ↑	Zhou et al.	simulator	0.49	0.60	0.72	**0.83**	0.86	–
PPol classifier ↑	Chopra et al.	rollout	0.04	0.35	0.66	**0.97**	0.77	0.80
w/o HP indicators ↑	Chopra et al.	rollout	0.07	0.31	0.53	**0.94**	0.68	0.72

Results. Every measure built on features disjoint from HP places TO farthest from the human value and both persona-conditioned methods closer to it (Table 14). Across the 24 simulators, these measures correlate with the HP pass rate, with ρ from 0.35 to 0.75, mostly through this gap between TO and the other two methods. Pooled over simulators, rollouts HP flags score as less human-like than those it passes, with AUC from 0.73 to 0.75. Within a single simulator the difference is smaller, with mean AUC from 0.55 to 0.63. Training the n-gram and Biber classifiers against all three methods instead of TO alone keeps TO lowest, with AUC of 0.64 and 0.70.

Passing HP does not make a rollout indistinguishable from a person, and the measures that compare simulators with people directly show a large remaining distance. Two halves of the human set reach a MAUVE of 0.95, while the methods reach at most 0.11. A BERT classifier tells TPS from people with 88% balanced accuracy, against 51% for two halves of the human set. The measures that share HP's behavioral statistics also agree, including a PPol classifier that omits HP's own indicators; we list them for completeness.

E Additional Results on User Simulation

E.1 Model-Level Comparison of TO, T+P, and TPS

Figure 13 and Table 34 give the model-level rates behind Section 4.3 for every user LLM evaluated under all three methods on both domains: 30 configurations, one per checkpoint. Rates add retail and airline counts within each model before division; means give models equal weight.

E.2 Ablations of the Two-Module Design

We examine the roles, inputs, and call structure of the two modules on the 114 retail tasks with seven large-scale and seven small/mid-scale user LLMs, a subset of the 15 in Figure 3 (OLMo3.1-32B omitted). The evaluation uses three personas per task, each simulated once; methods without a persona are run three times per task. Rates are percentages, and means give each available user LLM equal weight within each method. TPS uses the same full-panel reference throughout these ablations and the task-preparation comparisons below. Bold marks the best method within each comparable set; parentheses show changes from TPS in percentage points, omitted when coverage differs.

E.2.1 Effect of the Goal Tracker

To isolate the Goal Tracker, we remove the Persona Realizer (Goal Tracker only): the Goal Tracker keeps tracking the compiled scenario, no disposition is drawn and no linguistic style is given, and

the utterance is written from the current wants alone. Table 15 compares this variant with TO, T+P, and TPS on the 114 retail tasks, and Table 16 gives the per-model rates.

In both groups, the Goal Tracker alone raises TF and HP over TO and has the highest TF among the four methods, but persona identification remains near chance. Adding the Persona Realizer lowers TF while improving HP and joint pass and making the assigned persona identifiable. The two modules thus serve complementary roles: the Goal Tracker supports task fidelity, while the Persona Realizer preserves persona-specific behavior.

Table 15: Mean pass rates for the Goal Tracker ablation. ID: persona identification accuracy with Full profiles (Section 4.3; chance 33.3%).

	Large-scale (7)				Small/mid-scale (7)			
Method	TF	HP	Joint	ID	TF	HP	Joint	ID
TO	88.4	63.8	57.4	33.2	71.5	43.3	33.2	34.3
T+P	84.4	89.4	76.6	**68.4**	64.0	76.3	49.8	**65.8**
TPS	85.6	**94.3**	**81.8**	65.9	78.2	**92.0**	**73.3**	60.4
Goal Tracker only	**90.2** (+4.6)	86.5 (−7.8)	79.1 (−2.7)	33.8 (−32.1)	**83.9** (+5.7)	82.2 (−9.8)	70.2 (−3.1)	33.1 (−27.3)

Table 16: Per-model pass rates for the Goal Tracker ablation. Tracker only: TPS without the Persona Realizer.

	TF				HP				Joint			
User LLM	TO	T+P	TPS	Tracker only	TO	T+P	TPS	Tracker only	TO	T+P	TPS	Tracker only
DeepSeek V3.2	**90.6**	85.7	83.6	87.7	68.4	88.6	**95.3**	93.9	63.5	76.6	80.7	**83.6**
DeepSeek V4 Flash	**91.8**	90.9	88.6	91.5	65.2	90.4	**95.3**	75.1	59.9	81.9	**85.7**	70.8
DeepSeek V4 Pro	81.9	82.2	87.4	**91.5**	53.2	83.9	**98.8**	86.8	45.3	71.1	**86.8**	81.0
GPT-5.6-Terra	89.2	90.1	88.3	**91.5**	90.6	**96.8**	93.6	94.7	83.6	**88.3**	83.3	86.8
Kimi K2.5	90.4	84.8	89.5	**92.7**	40.6	88.6	**93.6**	89.8	36.5	75.7	**84.2**	83.9
Kimi K2.6	91.2	85.7	91.5	**91.8**	64.9	**93.0**	**93.0**	89.5	59.9	81.0	**86.0**	82.7
Mistral Large 3	83.6	71.6	70.5	**84.8**	63.7	84.8	**90.6**	75.4	53.2	61.4	**65.8**	65.2
Large-scale mean	88.4	84.4	85.6	**90.2**	63.8	89.4	**94.3**	86.5	57.4	76.6	**81.8**	79.1
GLM4.7 Flash	69.9	59.1	75.4	**76.6**	72.2	86.5	**94.4**	91.5	50.6	52.6	**71.6**	71.3
GPT-OSS 20B	71.9	59.4	81.3	**86.5**	30.7	64.6	**91.8**	74.6	22.8	36.8	**75.4**	65.5
Gemma 3 27B	62.0	62.6	66.7	**77.5**	53.5	94.2	**95.6**	88.3	36.0	59.1	65.2	**70.2**
Llama 3.3 70B	62.3	62.6	81.0	**83.3**	16.7	77.8	**88.3**	64.6	12.9	50.3	**76.0**	55.6
Nemotron3.5-30B-A3B	76.9	60.8	85.1	**88.2**	21.9	58.8	**89.8**	82.7	17.3	36.8	**77.8**	73.7
Qwen3.5-4B	67.5	55.6	69.9	**83.6**	33.3	69.6	**93.3**	83.6	25.1	39.8	66.1	**71.9**
Qwen3.8-27B	90.1	87.7	88.0	**91.5**	74.9	82.7	**90.9**	89.8	67.8	73.1	80.7	**83.3**
Small/mid-scale mean	71.5	64.0	78.2	**83.9**	43.3	76.3	**92.0**	82.2	33.2	49.8	**73.3**	70.2
All 14 mean	79.9	74.2	81.9	**87.1**	53.6	82.9	**93.2**	84.3	45.3	63.2	**77.5**	74.7

E.2.2 Inputs of the Persona Realizer

The Persona Realizer receives two inputs from the persona: the behavioral dispositions and the linguistic style. We ablate each input separately. In *Behavioral dispositions only*, the linguistic style is not given; in *Linguistic style only*, no behavioral disposition is drawn. The Goal Tracker and the realizer prompt are otherwise identical to TPS. The variants target the same three personas per task as the TPS reference. Table 17 reports the means and Table 18 the per-model rates.

Giving the Persona Realizer less of the persona raises TF in both groups. With one input removed, TF is 1.1 and 1.8 points above TPS for large-scale user LLMs and 3.1 and 1.9 points above it for small/mid-scale ones, and with no persona at all (Goal Tracker only) it is higher still, at 90.2 and 83.9. Each input makes the user state its request less directly, and each one alone already costs some task fidelity.

HP moves the other way. For small/mid-scale user LLMs, TPS with both inputs has the highest HP (92.0); removing either input lowers it by about two points (90.0 and 89.9), and removing both

lowers it to 82.2. Large-scale user LLMs hold HP within a point under either ablation (93.5 and 93.7 against 94.3), so for them the two inputs substitute for each other. Read along the strength of the persona intervention, from no persona through one input to both, TF falls step by step in both groups while HP rises step by step for small/mid-scale user LLMs and saturates after the first input for large-scale ones: a small residual trade-off within TPS, although TPS still retains higher TF than T+P in both groups(Section 4.3).

Table 17: Mean pass rates for the Persona Realizer input ablation.

	Large-scale (7)			Small/mid-scale (7)		
Method	TF	HP	Joint	TF	HP	Joint
TO	88.4	63.8	57.4	71.5	43.3	33.2
Goal Tracker only	90.2 (+4.6)	86.5 (−7.8)	79.1 (−2.7)	83.9 (+5.7)	82.2 (−9.8)	70.2 (−3.1)
TPS	85.6	94.3	81.8	78.2	92.0	73.3
Behavioral dispositions only	86.7 (+1.1)	93.5 (−0.8)	81.9 (+0.1)	81.3 (+3.1)	90.0 (−2.0)	74.0 (+0.7)
Linguistic style only	87.4 (+1.8)	93.7 (−0.6)	82.3 (+0.5)	80.1 (+1.9)	89.9 (−2.1)	73.2 (−0.1)

Table 18: Per-model pass rates for the Persona Realizer input ablation. Tracker only: TPS without the Persona Realizer, as in Table 16; Disp.: behavioral dispositions only; Style: linguistic style only.

	TF					HP					Joint				
User LLM	TO	Tracker only	TPS	Disp.	Style	TO	Tracker only	TPS	Disp.	Style	TO	Tracker only	TPS	Disp.	Style
Large-scale															
DeepSeek V3.2	**90.6**	87.7	83.6	85.7	86.3	68.4	93.9	95.3	95.6	**96.2**	63.5	83.6	80.7	83.0	**83.9**
DeepSeek V4 Flash	**91.8**	91.5	88.6	89.7	90.6	65.2	75.1	**95.3**	93.9	93.9	59.9	70.8	**85.7**	85.2	85.5
DeepSeek V4 Pro	81.9	**91.5**	87.4	86.0	89.5	53.2	86.8	**98.8**	96.2	94.4	45.3	81.0	**86.8**	83.6	84.5
GPT-5.6-Terra	89.2	**91.5**	88.3	87.4	88.9	90.6	**94.7**	93.6	92.4	94.2	83.6	**86.8**	83.3	81.9	83.9
Kimi K2.5	90.4	**92.7**	89.5	86.8	89.4	40.6	89.8	**93.6**	92.3	91.9	36.5	83.9	**84.2**	80.4	83.0
Kimi K2.6	91.2	91.8	91.5	**92.1**	91.1	64.9	89.5	**93.0**	92.1	92.7	59.9	82.7	**86.0**	84.8	83.8
Mistral Large 3	83.6	**84.8**	70.5	78.9	75.7	63.7	75.4	90.6	91.8	**92.4**	53.2	65.2	65.8	**74.3**	71.6
Large-scale mean	88.4	**90.2**	85.6	86.7	87.4	63.8	86.5	**94.3**	93.5	93.7	57.4	79.1	81.8	81.9	**82.3**
Small/mid-scale															
GLM4.7 Flash	69.9	**76.6**	75.4	**76.6**	75.4	72.2	91.5	94.4	**95.6**	90.6	50.6	71.3	71.6	**73.4**	69.6
GPT-OSS 20B	71.9	**86.5**	81.3	81.9	82.7	30.7	74.6	**91.8**	90.4	**91.8**	22.8	65.5	75.4	74.3	**76.0**
Gemma 3 27B	62.0	**77.5**	66.7	71.3	74.3	53.5	88.3	**95.6**	94.2	92.7	36.0	**70.2**	65.2	67.8	69.9
Llama 3.3 70B	62.3	**83.3**	81.0	82.5	78.9	16.7	64.6	**88.3**	83.3	86.8	12.9	55.6	**76.0**	71.1	70.5
Nemotron3.5-30B-A3B	76.9	**88.2**	85.1	87.7	83.3	21.9	82.7	**89.8**	89.5	82.5	17.3	73.7	77.8	**78.9**	71.1
Qwen3.5-4B	67.5	**83.6**	69.9	78.4	76.3	33.3	83.6	**93.3**	87.1	93.0	25.1	**71.9**	66.1	70.8	71.3
Qwen3.8-27B	90.1	**91.5**	88.0	90.6	89.8	74.9	89.8	90.9	89.8	**92.1**	67.8	83.3	80.7	81.9	**84.2**
Small/mid-scale mean	71.5	**83.9**	78.2	81.3	80.1	43.3	82.2	**92.0**	90.0	89.9	33.2	70.2	73.3	**74.0**	73.2

E.2.3 Call Structure of TPS

TPS applies the Goal Tracker's state update before the Persona Realizer generates the utterance. For this comparison we write its call structure as G/BL: one call for goal tracking (G), followed by one call in which the Persona Realizer chooses the move from the persona (B) and writes the utterance (L); a slash separates calls. We vary this structure in both directions on the 114 retail tasks. In *GBL*, one model reply performs goal tracking, selects the move, and writes the utterance; the state update is applied afterward for the next turn. In *G/B/L*, one call selects the move from the persona, the agent's message, and the current task state, and another realizes it as an utterance. Table 19 compares the three, and Table 20 gives the per-model rates.

We first compare G/B/L with G/BL. Choosing the behavioral response in one call and writing the utterance in another yields lower mean TF and joint pass in both groups, favoring their joint realization. GBL then asks whether goal tracking belongs there as well. Large-scale user LLMs gain in TF ($+2.7$) and joint pass ($+3.1$), suggesting that the appropriate call structure varies across user LLMs while task and persona roles remain distinct.

Table 19: Mean pass rates for the call-structure ablation (G/BL: TPS). G: goal tracking; B: choosing the move; L: writing the utterance; slashes separate calls.

	Large-scale (7)			Small/mid-scale (7)		
Method	TF	HP	Joint	TF	HP	Joint
G/BL (TPS)	85.6	94.3	81.8	**78.2**	92.0	**73.3**
GBL	**88.3** (+2.7)	**95.7** (+1.4)	**84.9** (+3.1)	74.7 (−3.5)	**92.1** (+0.1)	70.1 (−3.2)
G/B/L	79.7 (−5.9)	92.8 (−1.5)	75.2 (−6.6)	71.2 (−7.0)	87.4 (−4.6)	62.6 (−10.7)

Table 20: Per-model pass rates for the call-structure ablation (notation as in Table 19).

	TF			HP			Joint		
User LLM	G/BL (TPS)	GBL	G/B/L	G/BL (TPS)	GBL	G/B/L	G/BL (TPS)	GBL	G/B/L
DeepSeek V3.2	**83.6**	83.0	78.7	95.3	**97.7**	95.0	80.7	**81.6**	74.9
DeepSeek V4 Flash	88.6	**91.2**	82.5	95.3	**96.5**	87.1	85.7	**88.0**	74.9
DeepSeek V4 Pro	87.4	**89.2**	80.1	**98.8**	98.0	98.5	86.8	**87.4**	79.8
GPT-5.6-Terra	88.3	**90.1**	81.3	93.6	**96.5**	96.2	83.3	**87.4**	78.9
Kimi K2.5	89.5	**92.1**	80.4	93.6	**96.8**	89.8	84.2	**88.9**	74.6
Kimi K2.6	91.5	**92.1**	81.9	93.0	**94.2**	89.5	86.0	**86.5**	73.7
Mistral Large 3	70.5	**80.1**	72.8	90.6	90.4	**93.6**	65.8	**74.3**	69.9
Large-scale mean	85.6	**88.3**	79.7	94.3	**95.7**	92.8	81.8	**84.9**	75.2
GLM4.7 Flash	**75.4**	73.1	57.0	94.4	**96.5**	89.8	**71.6**	71.1	50.6
GPT-OSS 20B	**81.3**	80.7	77.8	91.8	**95.6**	88.9	75.4	**78.4**	70.2
Gemma 3 27B	**66.7**	53.8	57.6	95.6	79.2	**95.9**	**65.2**	45.3	55.8
Llama 3.3 70B	**81.0**	70.8	75.1	**88.3**	88.0	85.4	**76.0**	64.3	64.9
Nemotron3.5-30B-A3B	**85.1**	83.5	80.5	89.8	**93.1**	80.4	**77.8**	**77.8**	64.5
Qwen3.5-4B	69.9	68.1	**72.0**	93.3	**96.5**	82.5	**66.1**	65.2	61.7
Qwen3.8-27B	88.0	**92.7**	78.4	90.9	**95.9**	89.2	80.7	**88.9**	70.8
Small/mid-scale mean	**78.2**	74.7	71.2	92.0	**92.1**	87.4	**73.3**	70.1	62.6
All 14 mean	**81.9**	81.5	75.4	93.2	**93.9**	90.1	**77.5**	**77.5**	68.9

E.3 Effects of Input Preparation

We now examine the effects of task preparation and persona representation, separately from the two-module design, using the same seven large-scale and seven small/mid-scale user LLMs. In the following tables, rates are percentages, and means give each available user LLM equal weight within each method. Bold marks the best method within each comparison; parenthesized changes are in percentage points.

E.3.1 Task Preparation

TPS compiles each scenario into a fixed task structure before the dialogue (Appendix C.2). We examine this preparation from two sides: which model compiles the scenario, and whether the prompted methods also benefit from the compiled structure. Both preparation comparisons use the 114 retail tasks.

Compilation runs once per scenario, before any dialogue, so a task author can prepare it in advance with any model; our experiments use GPT-5.6-Terra. To see what this choice contributes, we let each user LLM compile its own scenario at its first turn with the same compilation prompt (TPS$^{\text{self}}$; Table 21, per-model rates in Table 23). The group means are similar for large-scale user LLMs, while shared offline compilation gives higher TF and joint pass for small/mid-scale user LLMs.

We also give TO and T+P the same compiled structure, rendered as prose in place of the original scenario (TO^{c}, T+P^{c}). Table 22 reports group means, and Table 24 the per-model rates.

Providing the compiled scenario does not consistently improve TO or T+P. TPS retains higher TF, HP, and joint pass than T+P^c in both groups, indicating that compiled input alone does not explain its advantage.

Table 21: Mean pass rates on 114 retail tasks with shared (TPS) or self-compiled task structures (TPS$^{\text{self}}$). Parentheses: change from TPS.

	Large-scale (7)			Small/mid-scale (7)		
Method	TF	HP	Joint	TF	HP	Joint
T+P	84.4	89.4	76.6	64.0	76.3	49.8
TPS	**85.6**	94.3	**81.8**	**78.2**	**92.0**	**73.3**
TPS$^{\text{self}}$	84.9 (−0.7)	**94.6** (+0.3)	80.7 (−1.1)	75.7 (−2.5)	90.1 (−1.9)	69.8 (−3.5)

Table 22: Mean pass rates on 114 retail tasks when TO and T+P receive the compiled scenario (TO^c, T+P^c). Parentheses: change from TO and T+P, respectively.

	Large-scale (7)			Small/mid-scale (7)		
Method	TF	HP	Joint	TF	HP	Joint
TO	**88.4**	63.8	57.4	71.5	43.3	33.2
TO^c	84.4 (−4.0)	61.1 (−2.7)	54.2 (−3.2)	70.9 (−0.6)	46.2 (+2.9)	35.5 (+2.3)
T+P	84.4	89.4	76.6	64.0	76.3	49.8
T+P^c	83.1 (−1.3)	88.5 (−0.9)	75.2 (−1.4)	61.6 (−2.4)	77.1 (+0.8)	49.2 (−0.6)
TPS	85.6	**94.3**	**81.8**	**78.2**	**92.0**	**73.3**

Table 23: Per-model pass rates on 114 retail tasks with shared (TPS) or self-compiled task structures (TPS$^{\text{self}}$), with T+P for reference.

	TF			HP			Joint		
User LLM	T+P	TPS	TPS$^{\text{self}}$	T+P	TPS	TPS$^{\text{self}}$	T+P	TPS	TPS$^{\text{self}}$
DeepSeek V3.2	**85.7**	83.6	79.2	88.6	95.3	**95.6**	76.6	**80.7**	76.9
DeepSeek V4 Flash	**90.9**	88.6	89.5	90.4	95.3	**96.2**	81.9	85.7	**86.3**
DeepSeek V4 Pro	82.2	**87.4**	85.7	83.9	**98.8**	**98.8**	71.1	**86.8**	85.1
GPT-5.6-Terra	**90.1**	88.3	87.4	**96.8**	93.6	93.3	**88.3**	83.3	82.2
Kimi K2.5	84.8	89.5	**90.9**	88.6	93.6	**93.9**	75.7	84.2	**85.7**
Kimi K2.6	85.7	**91.5**	89.5	93.0	93.0	**93.3**	81.0	**86.0**	83.3
Mistral Large 3	71.6	70.5	**71.9**	84.8	90.6	**90.9**	61.4	**65.8**	**65.8**
Large-scale mean	84.4	**85.6**	84.9	89.4	94.3	**94.6**	76.6	**81.8**	80.7
GLM4.7 Flash	59.1	**75.4**	74.9	86.5	**94.4**	90.1	52.6	**71.6**	69.0
GPT-OSS 20B	59.4	**81.3**	71.3	64.6	91.8	**93.6**	36.8	**75.4**	67.0
Gemma 3 27B	62.6	**66.7**	66.4	94.2	**95.6**	91.5	59.1	**65.2**	63.7
Llama 3.3 70B	62.6	**81.0**	75.4	77.8	**88.3**	81.3	50.3	**76.0**	65.5
Nemotron3.5-30B-A3B	60.8	**85.1**	83.8	58.8	89.8	**91.5**	36.8	**77.8**	76.9
Qwen3.5-4B	55.6	**69.9**	67.3	69.6	**93.3**	93.0	39.8	**66.1**	64.9
Qwen3.8-27B	87.7	88.0	**90.9**	82.7	**90.9**	89.8	73.1	80.7	**81.4**
Small/mid-scale mean	64.0	**78.2**	75.7	76.3	**92.0**	90.1	49.8	**73.3**	69.8
All 14 mean	74.2	**81.9**	80.3	82.9	**93.2**	92.3	63.2	**77.5**	75.3

Table 24: Per-model pass rates on 114 retail tasks when TO and T+P receive the compiled scenario (TO^c, T+P^c).

	TF					HP					Joint				
User LLM	TO	TO^c	T+P	T+P^c	TPS	TO	TO^c	T+P	T+P^c	TPS	TO	TO^c	T+P	T+P^c	TPS
DeepSeek V3.2	**90.6**	82.5	85.7	86.3	83.6	68.4	65.8	88.6	88.9	**95.3**	63.5	57.9	76.6	78.9	**80.7**
DeepSeek V4 Flash	**91.8**	88.3	90.9	86.8	88.6	65.2	61.7	90.4	86.0	**95.3**	59.9	56.4	81.9	75.4	**85.7**
DeepSeek V4 Pro	81.9	77.5	82.2	80.7	**87.4**	53.2	53.8	83.9	83.9	**98.8**	45.3	47.1	71.1	69.6	**86.8**
GPT-5.6-Terra	89.2	**90.4**	90.1	86.8	88.3	90.6	90.1	96.8	**97.1**	93.6	83.6	81.3	**88.3**	85.4	83.3
Kimi K2.5	**90.4**	84.2	84.8	84.2	89.5	40.6	41.2	88.6	86.3	**93.6**	36.5	36.5	75.7	73.1	**84.2**
Kimi K2.6	91.2	89.5	85.7	88.0	**91.5**	64.9	61.4	**93.0**	92.4	**93.0**	59.9	56.4	81.0	82.7	**86.0**
Mistral Large 3	**83.6**	78.4	71.6	69.0	70.5	63.7	53.8	84.8	85.1	**90.6**	53.2	43.9	61.4	61.1	**65.8**
Large-scale mean	**88.4**	84.4	84.4	83.1	85.6	63.8	61.1	89.4	88.5	**94.3**	57.4	54.2	76.6	75.2	**81.8**
GLM4.7 Flash	69.9	65.5	59.1	53.2	**75.4**	72.2	71.1	86.5	85.4	**94.4**	50.6	49.4	52.6	44.7	**71.6**
GPT-OSS 20B	71.9	75.1	59.4	61.4	**81.3**	30.7	42.4	64.6	70.2	**91.8**	22.8	34.0	36.8	43.3	**75.4**
Gemma 3 27B	62.0	**66.7**	62.6	59.6	**66.7**	53.5	59.6	94.2	94.2	**95.6**	36.0	43.6	59.1	57.0	**65.2**
Llama 3.3 70B	62.3	57.6	62.6	58.2	**81.0**	16.7	17.5	77.8	78.7	**88.3**	12.9	10.2	50.3	48.5	**76.0**
Nemotron3.5-30B-A3B	76.9	72.8	60.8	56.4	**85.1**	21.9	23.4	58.8	57.6	**89.8**	17.3	17.8	36.8	34.5	**77.8**
Qwen3.5-4B	67.5	**70.8**	55.6	58.2	69.9	33.3	38.6	69.6	69.3	**93.3**	25.1	29.2	39.8	44.2	**66.1**
Qwen3.8-27B	**90.1**	88.0	87.7	84.2	88.0	74.9	71.1	82.7	84.5	**90.9**	67.8	64.0	73.1	72.2	**80.7**
Small/mid-scale mean	71.5	70.9	64.0	61.6	**78.2**	43.3	46.2	76.3	77.1	**92.0**	33.2	35.5	49.8	49.2	**73.3**
All 14 mean	79.9	77.6	74.2	72.4	**81.9**	53.6	53.7	82.9	82.8	**93.2**	45.3	44.8	63.2	62.2	**77.5**

E.3.2 Persona Representation in T+P

T+P and TPS receive the persona profiles built by our extraction (Section 4.1). To see how this extraction shapes T+P, we give T+P the same person's dialogue in less processed form: the dialogue itself (T+P^d). The check runs on 40 of the 114 retail tasks. Table 25 reports group means, and Table 26 the per-model rates.

The extracted profile gives T+P higher HP and joint pass than the original dialogue in both groups, while the dialogue yields higher TF for small/mid-scale user LLMs.

Table 25: Mean pass rates on 40 retail tasks when T+P receives the person's dialogue (T+P^d) instead of the extracted profile. Parentheses: change from T+P.

	Large-scale (7)			Small/mid-scale (7)		
Method	TF	HP	Joint	TF	HP	Joint
T+P	84.8	**90.6**	**77.9**	62.6	**74.0**	**48.2**
T+P^d	**84.9** (+0.1)	83.2 (−7.4)	71.1 (−6.8)	**69.0** (+6.4)	65.7 (−8.3)	47.6 (−0.6)

Table 26: Per-model pass rates on 40 retail tasks when T+P receives the person's dialogue (T+P^d) instead of the extracted profile.

	TF		HP		Joint	
User LLM	T+P	T+P^d	T+P	T+P^d	T+P	T+P^d
DeepSeek V3.2	**84.2**	83.3	89.2	**91.7**	76.7	**77.5**
DeepSeek V4 Flash	**89.2**	76.7	**94.2**	65.0	**84.2**	50.8
DeepSeek V4 Pro	86.7	**90.0**	**90.0**	84.2	**80.8**	75.8
GPT-5.6-Terra	86.7	**87.5**	**98.3**	97.5	**85.8**	85.0
Kimi K2.5	**86.7**	83.3	**87.5**	71.7	**75.8**	58.3
Kimi K2.6	**87.5**	**87.5**	**89.2**	88.3	**79.2**	76.7
Mistral Large 3	72.5	**85.8**	**85.8**	84.2	62.5	**73.3**
Large-scale mean	84.8	**84.9**	**90.6**	83.2	**77.9**	71.1
GLM4.7 Flash	59.2	**62.5**	**80.8**	68.3	**49.2**	48.3
GPT-OSS 20B	56.7	**70.0**	**65.0**	47.5	34.2	**35.8**
Gemma 3 27B	65.0	**67.5**	91.7	**99.2**	60.8	**66.7**
Llama 3.3 70B	63.3	**65.0**	**75.0**	67.5	**50.0**	44.2
Nemotron3.5-30B-A3B	55.0	**58.3**	**52.5**	37.5	**30.0**	23.3
Qwen3.5-4B	51.7	**71.7**	**70.8**	55.0	**39.2**	**39.2**
Qwen3.8-27B	87.5	**88.3**	82.5	**85.0**	74.2	**75.8**
Small/mid-scale mean	62.6	**69.0**	**74.0**	65.7	**48.2**	47.6
All 14 mean	73.7	**77.0**	**82.3**	74.5	**63.0**	59.3

E.4 Comparison with Existing User Simulation Methods

Table 27 compares TPS with four existing methods on the 114 retail and 50 airline tasks, with Qwen3 8B as the user LLM and DeepSeek V4 Flash as the agent. Methods with a persona (T+P, TPS, PPol) use three personas per task, each simulated once; the others run three times per task.

- **UGST** (Mehri et al., 2026b) tracks the state of the user goal at each turn and feeds it back to the simulator; we use its inference-time stage.
- **NCUser** (Shim et al., 2026) injects non-collaborative behaviors into user turns; we use *fragment* (incomplete utterances), *tangential* (off-topic remarks), and *emotional* (frustration toward the agent), and omit *unavailable* service requests, which add new goals to the assigned task.
- **PPol** (Chopra et al., 2026) discovers user behaviors with evolutionary program search and turns them into role-play policies; since the policy is evolved per user LLM, we evolve and run it on Qwen3 8B.
- **OSim** (Zhou et al., 2026a) is an 8B model trained to simulate human behavior; we run it with TO, T+P, and TPS.

Results. TPS has the highest TF and joint pass in both domains, and the highest HP among methods run on Qwen3 8B, improving all three over TO (joint pass $+30.1$ pp in retail and $+30.7$ pp in airline). UGST, which targets task fidelity, also raises TF and HP over TO, but stays below TPS on all three measures. Methods that target human-likeness mostly raise HP but do not carry this gain over to task fidelity: PPol raises HP by 13.8 and 16.7 pp while changing TF by $+2.0$ pp in retail and -7.3 pp in airline, and NCUser's fragment and emotional behaviors lower TF in both domains, while its tangential behavior lowers both TF and HP. As a result, their joint-pass gains remain small or negative. OSim, trained on large-scale human behavior, improves both TF and HP over Qwen3 8B, which shares its base model, even under TO without a persona, and has the highest joint pass among existing methods in retail. This suggests that training on human behavior is a promising way to obtain human-plausible users without sacrificing task fidelity. TPS is complementary to such training: applied to OSim, it gives the highest HP in the table and raises joint pass over OSim with TO in both domains.

Table 27: Pass rates of TPS and existing user simulation methods. Parentheses: change from TO with Qwen3 8B. Bold: best per column.

			Retail			Airline		
	Method	User LLM	TF	HP	Joint	TF	HP	Joint
	TO	Qwen3 8B	44.2	60.2	30.7	41.3	49.3	24.0
Inference-time	T+P	Qwen3 8B	52.6 (+8.4)	75.7 (+15.5)	44.4 (+13.7)	45.3 (+4.0)	70.0 (+20.7)	31.3 (+7.3)
	TPS	Qwen3 8B	**68.4** (+24.2)	84.5 (+24.3)	**60.8** (+30.1)	**62.7** (+21.4)	84.0 (+34.7)	**54.7** (+30.7)
	UGST	Qwen3 8B	62.0 (+17.8)	77.2 (+17.0)	47.7 (+17.0)	53.3 (+12.0)	67.3 (+18.0)	37.3 (+13.3)
	NCUser (fragment)	Qwen3 8B	38.0 (−6.2)	81.9 (+21.7)	33.3 (+2.6)	30.0 (−11.3)	74.7 (+25.4)	22.0 (−2.0)
	NCUser (tangential)	Qwen3 8B	38.6 (−5.6)	48.5 (−11.7)	20.5 (−10.2)	38.7 (−2.6)	37.3 (−12.0)	16.7 (−7.3)
	NCUser (emotional)	Qwen3 8B	37.2 (−7.0)	71.5 (+11.3)	28.4 (−2.3)	34.8 (−6.5)	62.1 (+12.8)	20.3 (−3.7)
	PPol	Qwen3 8B	46.2 (+2.0)	74.0 (+13.8)	36.3 (+5.6)	34.0 (−7.3)	66.0 (+16.7)	26.7 (+2.7)
Model-based	TO	OSim 8B	64.9 (+20.7)	77.8 (+17.6)	52.6 (+21.9)	53.3 (+12.0)	63.3 (+14.0)	32.7 (+8.7)
	T+P	OSim 8B	50.9 (+6.7)	86.0 (+25.8)	43.9 (+13.2)	47.3 (+6.0)	78.7 (+29.4)	36.0 (+12.0)
	TPS	OSim 8B	59.1 (+14.9)	**91.5** (+31.3)	55.0 (+24.3)	57.3 (+16.0)	**88.7** (+39.4)	50.0 (+26.0)

E.5 Details on Persona Identifiability

Section 4.3 asks whether the persona assigned to a simulated user can be recovered from the dialogue. Each task has three persona profiles, each extracted from a different human dialogue on that task (Section 4.1). For every simulated dialogue, an LLM judge (GPT-5.6-Terra) reads the dialogue together with the three profiles of its task and chooses the profile of the simulated user. Because the three users pursue the same task, the profiles differ only in behavioral dispositions and linguistic style. We provide profiles containing behavioral dispositions alone (Behav.-only) or with linguistic style (Full). TO receives no persona, so each TO dialogue is scored against one of its task's profiles, which places TO at chance by construction. As a reference, each human dialogue is scored against the profile extracted from it.

We run this on the 114 retail tasks, using the rollouts of the 30 user LLMs under each method (30,778 dialogues) and the 342 human dialogues. We validate the judge with human raters, who performed the same three-way choice with Full profiles without seeing the judge's choice. On 500 sampled dialogues, human and judge choices agree with Cohen's $\kappa = 0.62$. Refer to Appendix H.7 for the judge prompts.

E.6 Human Evaluation of the Persona–Task Fidelity Trade-off

We complement the automatic evaluation in Section 3.3 with human judgments of task fidelity and persona expression in TO and T+P dialogues.

Setup. We selected 200 matched model–task pairs from retail settings. These selected pairs represent trajectories conducted under the exact same task, differing only in the user simulator condition. We sampled evenly across models, taking one TO dialogue and one T+P dialogue per matched model–task pair. Eight raters completed judgments for the resulting 400 dialogues. Raters judged whether the simulated customer followed the assigned task based on the task description and dialogue, without seeing the persona. They then read the persona description and rated its expression in the dialogue as *clear*, *some*, *not expressed*, or *no opportunity to express*. The same questions were used for TO dialogues, with the corresponding persona provided for the expression judgment.

Evaluation criterion. We count dialogues in which a human-judged TF violation co-occurs with persona expression, treating *clear* and *some* as expression. This records the presence of both in a dialogue, without attributing the violation to the persona.

Results. Persona expression and a human-judged TF violation co-occurred in 79 of 200 T+P dialogues (39.5%) and 22 of 200 TO dialogues (11.0%; Table 28). In this sample, the combination was more frequent under T+P. These rates describe the selected cases rather than the prevalence across all TO and T+P dialogues.

Prompt	n	Persona expression + TF violation	Rate
T+P	200	79	39.5%
TO	200	22	11.0%

Table 28: Persona expression with a human-judged TF violation.

E.7 Pairwise Human Evaluation of HP

We check whether dialogues that violate HP also read as less human to raters.

Setup. Raters judged 150 pairs of simulated retail dialogues from the TO, T+P, and TPS conditions, balanced across pairs in which both dialogues pass HP, both violate it, or one of each. The two dialogues in a pair share a task, and the agent's turns are hidden. For each pair, raters answered which customer was a real person (A only, B only, both, or neither). These pairs were mixed with pairs containing in-domain human dialogues, so either customer could be a real person.

Results. Each answer labels both dialogues in a pair as human or not. When HP flags a violation, raters labeled the dialogue non-human in 93% (139/150); when HP passes a dialogue, raters labeled it human in 58% (87/150) (Table 29). Raters thus nearly always agree with a violation, while passing HP does not ensure that a dialogue reads as human, consistent with a boundary that flags only behavior people do not show. These rates describe the sampled pairs rather than the rollout population. We also use these judgments to choose the confidence level of the boundary (Appendix D.2): among the dialogues flagged at 95% but not at 99%, raters labeled 87.2% non-human, so the stricter 95% boundary mostly flags behavior that raters also find non-human.

HP verdict	Labeled human	Labeled non-human	Agreement
Pass	87	63	58%
Violation	11	139	93%

Table 29: Rater labels on the 300 simulated dialogues of the 150 pairs. Agreement is the share labeled human for passing dialogues and non-human for violating ones.

E.8 Human Evaluation of TF, HP, and Benchmark Outcomes

We assess whether the automatic TF and HP verdicts agree with human judgments and whether raters accept benchmark outcomes as evidence of agent capability.

Setup. We sampled 80 TO dialogues, selected across the four automatic TF–HP quadrants. Raters read the assigned scenario and full dialogue and answered three questions in order, each with *yes*, *no*, or *unsure*. First, they judged whether the simulated user remained faithful to the scenario (TF). Second, they judged whether its behavior was plausible for a real user (HP). Third, after both answers were recorded, the τ^2-bench outcome was revealed, and the question depended on it. If the benchmark scored the task a success (reward $= 1$), raters were asked "*Based on this success, is it reasonable to judge that this agent could also handle the same task well for a real customer?*" If it scored the task a failure (reward $= 0$), they were asked "*Based on this failure, is it reasonable to judge that this agent lacks the ability to complete the assigned task?*"

Agreement with human judgments. Agreement between human judgments and the automatic verdicts was 78.8% (63/80) for TF and 95.0% (76/80) for HP (Table 30). Both checks broadly align with human judgments, with higher agreement for HP. These rates describe the selected sample rather than the rollout population.

Interpreting benchmark outcomes. Among the 55 dialogues with an automatic TF or HP violation, raters answered *no* to the third question in 41 cases (74.5%): they did not take a success as evidence that the agent would handle the task for a real user, or a failure as evidence that the agent

lacks the ability to complete it. These judgments support considering TF and HP when interpreting benchmark successes and failures, consistent with the joint interpretation in Section 2.

Measure	Dialogues	Rate
TF: human–automatic agreement	80	78.8%
HP: human–automatic agreement	80	95.0%
Outcome interpretation: *No*	55	41/55 (74.5%)

Table 30: Human judgments on the 80 retained TO dialogues. The outcome row covers the 55 dialogues with an automatic TF or HP violation.

F Details on Agent Evaluation

F.1 Score Decomposition and Variation

These analyses use the ten agents and eight user LLMs of Section 5; the setup is in Appendix B. Rankings always use the original benchmark success rates, without filtering by TF or HP.

Score decomposition. We use a descriptive two-way ANOVA decomposition, treating agent LLM and user LLM as the two factors (Figure 7(a)). For each simulation method, we first average success rates across user LLMs for each agent and across agents for each user LLM. Each mean's deviation from the overall mean defines its corresponding main effect. Adding the two effects to the overall mean gives an additive fitted score for each agent–user pair; the residual is the observed score minus this fitted score.

For each method, let $S_{a,u}$ be the success rate of agent a with user LLM u, in percentage points. Each cell receives equal weight. With $A = 10$ agents, $U = 8$ user LLMs, and grand mean $\bar{S}$, define

$$\alpha_a = \bar{S}_{a,\cdot} - \bar{S}, \qquad \beta_u = \bar{S}_{\cdot,u} - \bar{S}, \qquad E_{a,u} = S_{a,u} - \bar{S} - \alpha_a - \beta_u.$$

Here, $\bar{S}_{a,\cdot}$ averages over user LLMs and $\bar{S}_{\cdot,u}$ averages over agents. Writing $\mathrm{SS}_{\mathrm{total}} = \sum_{a,u}(S_{a,u} - \bar{S})^2$, the agent, user-LLM, and residual shares, respectively, are

$$100\frac{U\sum_a \alpha_a^2}{\mathrm{SS}_{\mathrm{total}}}, \qquad 100\frac{A\sum_u \beta_u^2}{\mathrm{SS}_{\mathrm{total}}}, \qquad 100\frac{\sum_{a,u} E_{a,u}^2}{\mathrm{SS}_{\mathrm{total}}}.$$

These shares sum to 100% and describe observed score variation rather than causal effects. The residual captures variation not explained by the additive agent and user-LLM effects, including interactions and sampling variability.

Table 31: **Rollout diagnostics and ranking concordance.** Successes with HP violations use all scored rollouts as the denominator, as in Figure 6. Agent share and remainder SD describe 80 success rates per method; W is tie-corrected concordance across eight user LLMs. Conditions receive equal weight.

Simulator	HP-violating success (/100)	Agent share (%)	Remainder SD (pp)	Ranking W
TO	21.1	33.0	8.2	0.692
T+P	7.2	46.7	6.5	0.790
TPS	**3.3**	**90.2**	**3.8**	**0.949**

Table 31 gives full-grid diagnostics. Remainder SD is $\sqrt{\sum_{a,u} E_{a,u}^2/(AU)}$. The corresponding agent-effect SDs are 10.9, 11.2, and 19.5 points, so the lower remainder does not simply collapse agent differences.

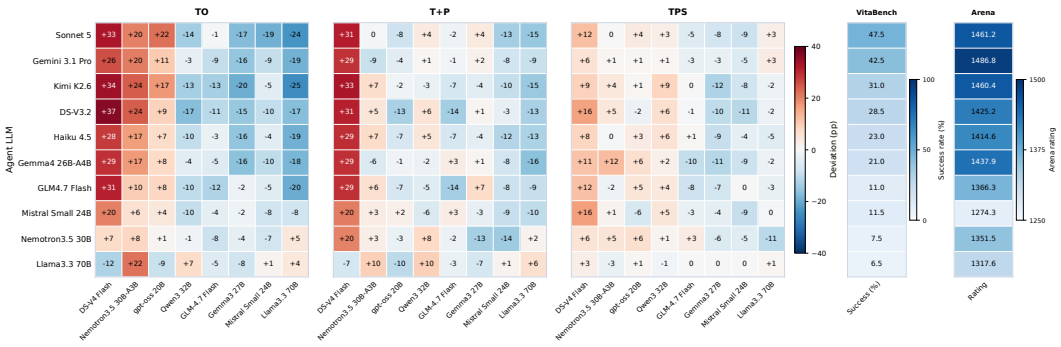


Figure 11: Within-agent success-rate deviations (pp); VitaBench and Arena scores use separate scales.

Within-agent score variation. For each agent, we compute the descriptive SD over the eight observed user LLMs, $s_a = \sqrt{\frac{1}{8}\sum_u (S_{a,u} - \bar{S}_{a,\cdot})^2}$. Averaging these SDs across the ten agents gives 14.8, 11.8, and 6.1 percentage points under TO, T+P, and TPS, respectively. Unlike remainder SD, this measure includes systematic score shifts associated with the user LLM. Figure 11 visualizes these within-agent score deviations with agents as rows and user LLMs as columns, ordered by mean success rate across all agents and methods. The three simulation methods share a color scale; VitaBench success rates and Arena ratings appear on the right with separate scales. Stronger color contrasts across user LLMs within an agent's row indicate greater score variation under that simulation method.

F.2 Outcome Diagnostics and Uncertainty

Figure 6 uses all scored rollouts as the denominator rather than normalizing successes and failures separately as in Figure 2. Mean failure rates in the multi-agent grid are 57.9%, 61.3%, and 46.4% under TO, T+P, and TPS; TF-violating failures account for 30.1%, 38.6%, and 21.9% of all scored rollouts. These rates are averaged equally across agent–user conditions.

Figure 6(b) uses 95% percentile intervals from 10,000 task-bootstrap draws. Each draw samples 114 task IDs with replacement, sharing task multiplicities across all methods and agent–user conditions. Condition-specific denominators are recomputed before averaging. Agent and user LLMs, assigned profiles, and observed rollout realizations remain fixed. These intervals concern task sampling, not repeated-rollout or model-population variance.

F.3 Ranking Robustness

Kendall's W uses average ranks and the standard denominator correction for ties. Figure 7(b) displays $W' = 1 - W$ only to express disagreement; the source statistic and Table 31 retain tie-corrected W. A reversed agent pair has both positive and negative signed score differences across user LLMs. Each pair counts once, whether one user or several users produce the exceptional ordering. Ties provide neither direction. Pairs with fewer than two non-tied outcomes are unresolved rather than stable; all 45 pairs are comparable here.

An agent pair's gap is the absolute value of its mean signed score difference across all combinations of the three simulation methods (TO, T+P, TPS) and eight user LLMs, so all methods use the same pair subsets. The main figure considers pairs with gaps above five percentage points; Table 32 reports all-pair results and cutoff sensitivity.

Figure 12 shows the agent rankings under each user LLM alongside VitaBench and Arena rankings, all computed within the same ten-agent panel. It retains the row and column order of Figure 11. For display, tied agents share a rank and subsequent ranks are skipped (e.g., 1, 2, 2, 4); the correlation and concordance statistics retain average ranks for ties.

Table 32: Agent pairs with at least one strict ordering reversal across user LLMs, divided by comparable agent pairs. Each pair counts once. All methods use the same gap strata; summary rows overlap the finer bins, and all 45 pairs are comparable for every method. The bold row is the one reported in Figure 7(b).

Mean score gap	TO	T+P	TPS
All pairs	26/45	23/45	8/45
$\leq$ 5 pp	7/7	7/7	6/7
5–10 pp	10/11	8/11	2/11
10–20 pp	7/12	6/12	0/12
$\geq$ 20 pp	2/15	2/15	0/15
>5 pp	**19/38**	**16/38**	**2/38**
$\geq$ 10 pp	9/27	8/27	0/27

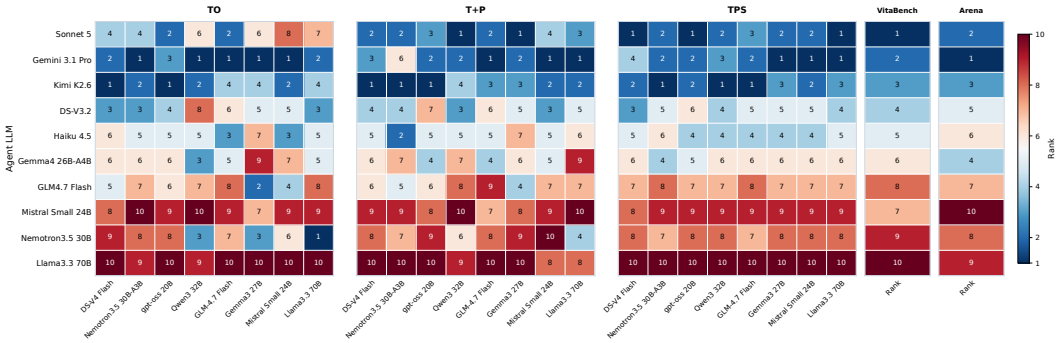


Figure 12: Agent rankings across user LLMs, VitaBench, and Arena (1 is best; ties share ranks).

F.4 Reference Evaluations

Table 33: Spearman correlation with each reference: mean [minimum, maximum] over eight user LLMs, not confidence intervals. Parentheses in headers give the number of agents.

Simulator	VitaBench (10)	Arena (10)
TO	0.63 [0.30, 0.92]	0.71 [0.38, 0.92]
T+P	0.83 [0.66, 0.93]	0.83 [0.63, 0.90]
TPS	0.92 [0.88, 0.96]	0.92 [0.86, 0.95]

VitaBench. We evaluate the same ten agent LLMs on 200 English VitaBench tasks: 100 delivery and 100 in-store tasks. GPT-4.1 serves as the user simulator, and GPT-5.6-Terra evaluates task success. Each agent's reference score is its success rate over the 200 tasks.

Arena. Arena is a crowdsourced platform that ranks LLMs using human preferences between responses from two anonymous models.[3] We use the ten agent LLMs' scores from its 13 September 2026 overall text leaderboard as reference scores.

For each simulation method and user LLM, we compute Spearman correlations between the ten agents' success rates and their scores in each reference, using average ranks for ties. Table 33 reports the mean and range across the eight user LLMs.

[3] https://arena.ai/leaderboard/text

Table 34: Combined retail and airline pass rates (%) for all 30 user LLMs, in the order of Figure 13. Bold: best of the three methods per metric; underline: better of T+P and TPS, only in TF.

	TF			HP			Joint		
User LLM	TO	T+P	TPS	TO	T+P	TPS	TO	T+P	TPS
GPT-5.6 Terra	**87.0**	86.4	85.8	92.1	**96.1**	92.7	82.7	**84.3**	79.5
GPT-5.6 Sol	**88.4**	86.4	85.8	90.9	**95.3**	89.2	81.5	**83.5**	76.4
Claude Sonnet 5	**84.6**	82.2	81.1	81.5	96.1	**96.5**	69.7	**79.8**	78.7
Claude Haiku 4.5	78.5	78.7	**79.9**	41.3	84.6	**90.7**	32.9	67.5	**73.8**
Gemini 3.1 Pro	**89.6**	83.7	**89.6**	77.2	90.9	**93.3**	69.7	75.8	**83.3**
Gemini 3.5 Flash Lite	77.2	63.8	**81.5**	87.8	**94.9**	93.3	68.5	61.6	**77.0**
DeepSeek V4 Pro	78.7	79.1	**82.9**	51.4	81.1	**98.6**	42.7	66.5	**82.3**
DeepSeek V4 Flash	**88.2**	87.0	85.4	63.4	88.0	**94.3**	56.3	76.4	**81.3**
DeepSeek V3.2	**87.6**	83.3	77.6	69.1	88.0	**95.7**	62.2	74.0	**75.0**
Kimi K2.6	87.2	82.9	**88.2**	62.2	89.6	**92.1**	54.7	75.2	**81.7**
Kimi K2.5	84.6	79.7	**84.8**	36.2	85.4	**93.5**	31.1	68.3	**79.5**
Mistral Large 3	**80.5**	69.3	66.7	58.9	82.3	**91.3**	49.8	57.9	**62.4**
Llama 3.3 70B	60.2	61.6	**77.0**	14.0	76.4	**89.6**	10.2	49.4	**72.0**
Qwen3 32B	66.9	67.5	**81.9**	34.1	80.3	**90.9**	22.2	56.3	**75.6**
OLMo 3.1 32B	**75.8**	64.8	71.3	33.3	**82.1**	79.5	26.4	54.5	**58.5**
Gemma 4 31B	**90.9**	81.1	84.3	90.9	**97.0**	96.3	**82.7**	79.3	81.5
Nemotron 3.5 30B-A3B	72.6	59.1	**83.3**	20.3	57.9	**89.6**	15.7	35.0	**76.4**
GLM-4.7 Flash	68.5	57.1	**70.9**	72.6	86.6	**93.9**	49.0	50.6	**67.3**
Qwen3.8 27B	**88.4**	82.7	85.0	69.7	80.3	**90.7**	62.8	67.3	**77.4**
Gemma 3 27B	**64.4**	59.8	63.2	55.1	93.5	**93.9**	38.2	56.1	**61.4**
Gemma 4 26B-A4B	**83.3**	80.1	78.0	92.9	**97.2**	96.5	77.6	**78.0**	76.2
Mistral Small 24B	**67.3**	41.5	57.9	**83.7**	75.0	80.5	**57.9**	34.3	49.0
GPT-OSS 20B	68.3	57.5	**80.3**	30.3	65.9	**91.1**	21.7	36.2	**73.8**
Qwen3 14B	68.5	64.4	**75.8**	42.9	71.3	**88.6**	31.1	45.9	**68.5**
Qwen3.5 9B	74.6	65.9	**78.0**	43.7	73.8	**94.5**	35.0	50.4	**75.8**
Qwen3 8B	43.4	50.4	**66.7**	56.8	74.0	**84.3**	28.7	40.4	**58.9**
Llama 3.1 8B	**38.5**	31.7	31.9	44.8	**80.9**	54.7	19.3	**26.6**	21.5
Ministral 8B	**25.2**	14.0	18.3	67.7	**72.6**	71.3	**19.5**	11.6	14.6
Qwen3.5 4B	**69.3**	53.9	**69.3**	32.3	69.9	**94.1**	24.2	38.6	**65.9**
Qwen3 4B	**64.4**	54.5	58.7	47.6	**81.5**	68.1	33.3	**46.5**	44.1
All 30 mean	73.4	67.0	**74.0**	58.2	82.9	**89.0**	45.2	57.6	**68.3**

G Extended Related Work

LLM-based user simulators LLM-based user simulators are increasingly used to play the user when dialogue systems and interactive agents are evaluated or trained (Ni et al., 2026; Barres et al., 2025; Kong et al., 2024). In these settings, the simulator does more than generate natural-language inputs: through its responses, it determines how the assigned task unfolds over multiple turns. This creates a fidelity problem when the simulator drops required requests, changes constraints, or otherwise departs from the original user goal as the conversation grows. Prior work has therefore introduced explicit mechanisms for tracking the user goal or dialogue state across turns (Dongre et al., 2025; Mehri et al., 2026b).

A separate concern is whether the behavior produced by the simulator resembles the behavior of people who would interact with the deployed system. Off-the-shelf LLMs tend to be more cooperative than human users, for example by stating their goals clearly or readily providing requested information, which can make the resulting interaction easier for the agent (Zhou et al., 2026b). Existing approaches address this issue by injecting non-cooperative or behaviorally diverse personas (Shim et al., 2026; Chopra et al., 2026), or by training user models on human conversations (Naous et al., 2026; Wu et al., 2026; Wang et al., 2026). These two lines of work target different aspects of simulation: the former fidelity-oriented work aims to preserve what the user is supposed to accomplish, whereas the latter aims to better model how people pursue those goals. We address both

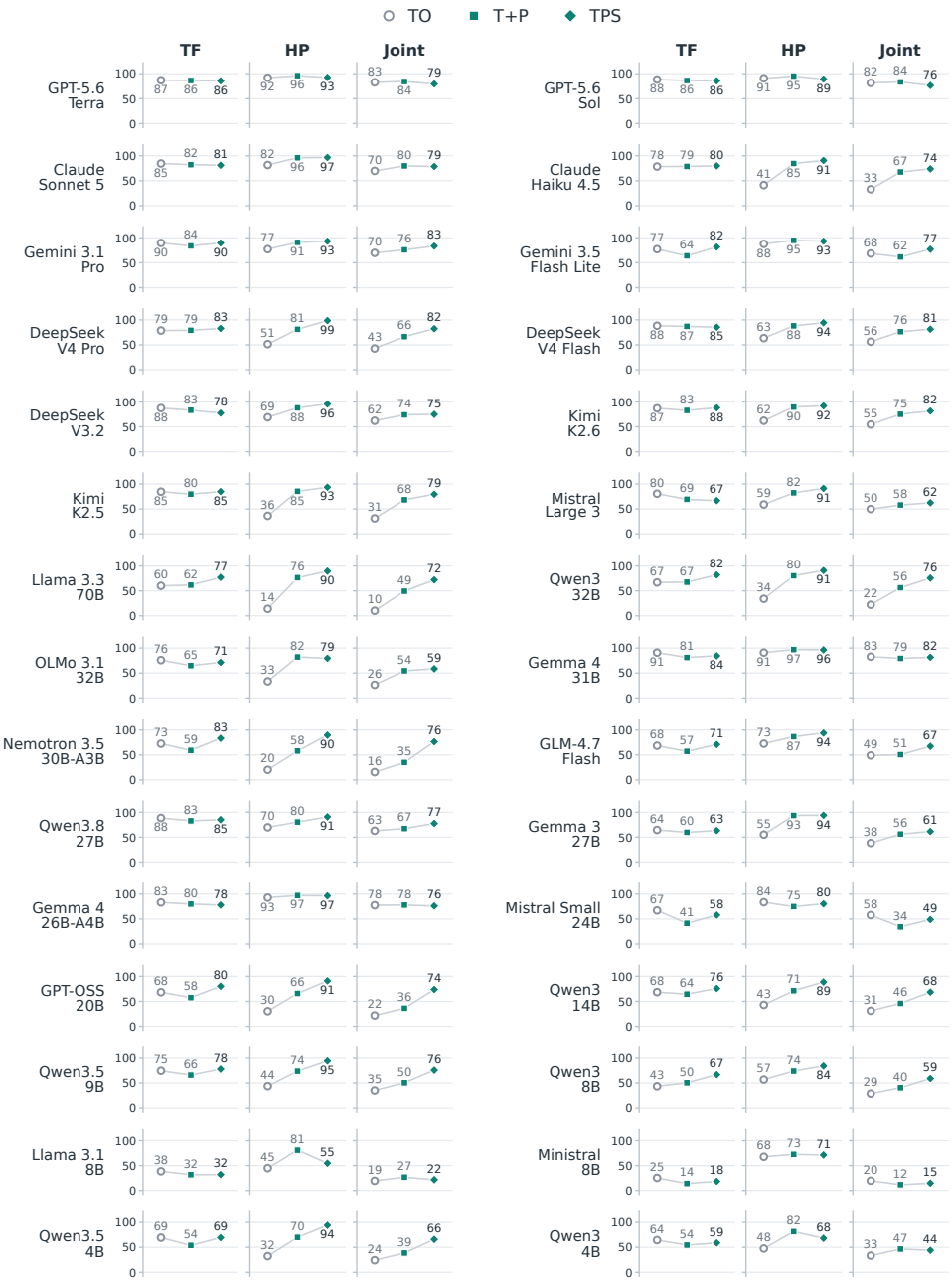


Figure 13: Combined retail and airline TF, HP, and joint pass rates for all 30 user LLMs, using the same domain-weighted rates as Table 34. Two models are shown per row.

problems within the same simulator by separating task-state tracking from persona-conditioned response generation.

Human-likeness of user simulators The realism of user simulators has been studied since statistical user simulation, but there remains no accepted criterion for how much resemblance to human behavior is sufficient (Schatzmann et al., 2006; Georgila et al., 2006; Pietquin & Hastie, 2013). This question is especially relevant for LLM-based simulators because human-likeness can refer to several different properties. A simulator may produce fluent, natural-looking language while following interaction strategies that differ from those of people, or it may match aggregate statistics without producing realistic individual conversations.

Recent work can be organized around three complementary notions of human-likeness. *Distinguishability* asks whether simulated users can be distinguished from real users (Ai & Litman, 2008; Hathidara et al., 2026; Meshi et al., 2026). *Similarity* measures how closely the simulator's language or interaction behavior matches human data (Ivey et al., 2026; Yoon et al., 2024; Mehri et al., 2026a). *Proxy validity* instead asks whether conclusions obtained with simulated users carry over when systems interact with people (Zhang & Balog, 2020; Seshadri et al., 2026; Suh et al., 2026). Our Human Plausibility verdict is most closely related to distinguishability, but adopts a narrower criterion: rather than asking whether simulated and human conversations are globally indistinguishable, it asks whether a single rollout exhibits behavior outside the calibrated range observed in human conversations. A passing verdict therefore indicates the absence of measured extreme behavior, rather than distributional equivalence to human users or validated transfer to real-user evaluation.

H Prompts and Examples

This appendix collects the prompts of the pipeline in the order they are used: persona extraction and annotation, scenario compilation, the user simulators (*task-only prompting*, TO; *task+persona prompting*, T+P; and *Task–Persona Separation*, TPS, following the terminology in Sections 3.1 and 4.3), and evaluation (the task-fidelity judge and persona identification), followed by two worked examples: from a human dialogue to a profile, and one turn of the simulator. Blue text marks the slots filled in per call. Fixed wording is printed once in the templates, and the worked examples give the filled values and recorded outputs. In each box, the turn-specific input is shaded and placed before the output guidance.

H.1 Persona Extraction and Annotation Prompts

We show the core prompts for contrastive extraction, category induction, and turn-level grounding (Section 4.1). Behavioral and linguistic categories are induced separately and subsequently consolidated. Shaded regions contain the inputs supplied for each call.

1. Contrastive extraction

```
You compare two conversations in which a customer talks to a support agent.  Both customers were given the
same task.  Each transcript is labelled with who the customer is:  a real person, or a language model
simulating a customer.  The agent is a language model in both.
Your job is to find what the customer in A does that the customer in B does not, and to keep only the part
of that difference that is PERSONALITY: how this person tends to behave and speak, independent of the task.
The brief is the task:  what the customer wants, facts they know or do not know, and any conditions or
scripts about how to carry it out ("if the agent refuses, mention X", "give the wrong zip code first, then
correct it").  Doing what the brief says is DOING THE TASK. It is not personality, and it is not something
to report, even when B does not do it.
Report an observation only when all of these hold:
  1.  A does it at some moment in the conversation, and you can quote it.
  2.  It is about how A behaves or speaks, not about what A is trying to obtain.
  3.  B either does something different at the same kind of moment, or that kind of
     moment never arises in B. Say which.  Do not treat "the moment never arose in B"
     as B behaving differently.
For each observation give:
  when          the kind of moment, stated so it could be recognised in any
                conversation:  "the agent asks for identifying details",
                "the agent explains what will happen next", "any turn".
                Always describe what the AGENT did in the message right before
                A's turn.  If A's turn is the first message of the conversation,
                write "start of conversation".  Never describe the moment by what
                A does.
  then_A        what A does at that moment, in plain words
  then_B        what B does at the same kind of moment, or exactly "no occasion"
  kind          "behavior" if it is about what A does or decides;
                "lexical" if it is about wording, length, punctuation, tone
  turn_A        the turn number in A where it shows most clearly
  quote_A       at most 80 characters copied verbatim from that turn of A
Report every distinct observation, not only the first.  If nothing qualifies, return an empty list.  Do not
invent a difference to have something to say.
```

1. Contrastive extraction (continued)

```
# The brief
{brief}
# Transcript A --- {a_who}
{a}
# Transcript B --- {b_who}
{b}
```

```
Return ONLY this JSON object:
{"observations": [{"when": "...", "then_A": "...", "then_B": "...", "kind": "...",
                   "turn_A": <int>, "quote_A": "..."}]}
Return the JSON now.
```

2a. Batch induction — behavioral categories

```
You are building a vocabulary of MOVE TYPES for customers in support chats. Below are descriptions of
moves a customer made, written independently by many annotators, one per line, with how often each was
written. The situation the move was made in has been deliberately removed.
Group the moves into types by WHAT KIND OF MOVE it is, regardless of what the agent had just asked. A type
must be a tendency a person could show at many different moments ("gives less than was asked", "keeps
pressing after a refusal"), not a situation ("when asked for an email"). Aim for between 8 and 16 types;
merge near-duplicates; do not invent a type no line supports. Include a type for plain compliance (does
what was asked, no more, no less) so that every move has a home.
```

```
• ({n}x) {description}
• ({n}x) {description}
• ...
```

```
Return ONLY this JSON:
{"types": [{"id": "<snake_case>", "name": "<short name>",
            "definition": "<one sentence: what the customer does, phrased so it can be
                           recognised in any situation>",
            "examples": ["<3 lines copied verbatim>", "...", "..."]}]}
Return the JSON now.
```

2b. Batch induction — linguistic categories

```
You are building a vocabulary of how customers WRITE in a support chat: wording, length, punctuation,
capitalisation, spelling, tone, register, greetings and sign-offs. Below are descriptions of one
customer's writing, one per line, written independently by many annotators.
Group them into distinct writing habits. Aim for between 10 and 30 habits. Merge wording differences;
keep real differences. Each habit must be something that could be read off a customer's messages alone.
```

```
• ({n}x) {description}
• ({n}x) {description}
• ...
```

```
Return ONLY this JSON:
{"habits": [{"id": "<snake_case>", "name": "<short name>",
             "definition": "<one sentence, in the form 'the customer ...'>",
             "examples": ["<3 lines copied verbatim>", "...", "..."]}]}
Return the JSON now.
```

3. Turn-level grounding

```
You annotate one conversation between a customer and a support agent, using a fixed vocabulary. The
customer was given a brief before the conversation. Work through the customer's turns in order.
Most of what a customer does is pursuing the goal their brief gives them, and how they pursue it is what
these actions describe. But some briefs go further and script a MOVE: a change of course to make if the
first thing cannot be had, a revision to make at a set point such as the agent's confirmation, a fact to
withhold or to state wrongly before correcting it, or a reaction to have when refused. A customer making
one of those scripted moves is following an instruction, not showing what they are like.
For every PERSON turn give:
  turn          its number
  situation     the id of the situation that the AGENT's message right before it belongs to, from
SITUATIONS. For the customer's first message use "conversation_opening". If nothing fits, use "none".
  actions       ids of what the customer does in this turn OF THEIR OWN ACCORD, chosen ONLY from the
actions listed under that situation. Include every action that clearly applies; zero or more. Empty when
the situation is "none" or has no actions listed.
  brief_actions ids from the same list for a SCRIPTED MOVE the customer makes in this turn, as described
above. This list is usually empty: only a handful of briefs script a move, and only at the one or two
turns where it happens.
  habits        ids from HABITS that are visible in the wording of this turn; zero or more.
An action goes in actions, not brief_actions, whenever the brief only says what the customer wants and the
action is about how they went about getting it -- how much they said at once, how precisely, whether they
greeted, confirmed, pushed back, asked why, or held something back. That the brief asked for the outcome
does not make pursuing it a scripted move. One turn can have both: a customer who makes the brief's
scripted change while also asking what it will cost has the first in brief_actions and the second in
actions. When you cannot tell, put it in actions.
```

```
# Brief
{brief}
# SITUATIONS, each with its actions
{vocab}
# HABITS
{habits}
# Conversation
{transcript}
```

```
Judge only from what is written. Do not infer intentions. Return ONLY this JSON object:
{"turns": [{"turn": <int>, "situation": "<id>", "actions": ["<id>", ...],
            "brief_actions": ["<id>", ...], "habits": ["<id>", ...]}, ...]}
with exactly one entry per PERSON turn, in order.
```

3. Turn-level grounding (continued)

```
Return the JSON now.
```

H.2 SCENARIO COMPILATION PROMPT

This is the prompt that turns a scenario into the fixed task structure of Section 4.2 (Appendix C.2). It is run once per scenario before any dialogue; the user message carries the scenario as the simulator sees it.

Scenario compilation — system message

```
You turn a customer's brief for a support chat into a scenario spec.  You see ONLY the brief.  The spec
must contain nothing the brief does not say.
A spec is an ORDERED LIST of items:  the separate things the customer came for, in the order they would
naturally raise them.  An item is a small TREE of rungs.  A rung is one thing the customer wants for that
item; the root rung is what they want at the outset, and every other rung replaces the rung it hangs off
when a named agent event happens.
  item.id       T1, T2, ...
  item.when     "" (the customer holds this item from the start), or "after T1" ONLY when the brief makes
this request wait for T1 to be settled ("only once ...  is done", "after that, ...").  "Also", "finally",
"at the end" are list order, not a wait.
  rung.id       a, b, c, ...
  rung.want     one want, written so it completes "you want ..." ("to exchange the keyboard for ...", "to
know the total you get back").  It must stand on its own:  carry over from the brief the object, its
qualifiers, the reason and any standing condition ("the more expensive of the two tablets", "only if it is
free", "but nothing else"), in the brief's own words and values.  Details the customer would give if asked
(a payment method, a reason, a preference among options) belong inside the want they serve; they are never
rungs of their own.  "∅" means the customer drops the item here.
  rung.from     the rung this one replaces; "" for the root.
  rung.when     what makes it replace its parent:
                  - the AGENT event, in the brief's words ("the agent tells you there is no ...", "the
agent asks you to confirm ...", "the agent says the price is more than 3000"), or
                  - "said":  the brief has the customer voice the parent and then move on by themselves
("say you want to think it over, then change your mind").
                Different agent answers with different follow-ups are sibling rungs under the same parent.
Rules
 • Every "if the agent says X, you want Y" is a rung under the want it replaces.  A fallback never becomes a
   separate item.
 • When the brief makes a follow-up depend on something the customer has to find out ("if the agent cannot
   guarantee ..."), asking about it is part of the parent want.
 • "First A, then B" is rungs when B replaces A as what is wanted, and list order when A and B are
   independent things the customer wants.
 • Things the customer wants outright are items with one rung.
 • Do NOT resolve anything:  never write which branch happens, never invent an id, a price, a stock fact, or
   an agent line the brief does not contain.  If the brief names a value, keep it verbatim; if it does not,
   do not supply one.
 • talk_rules:  how the customer talks and behaves, beside the spec and not in it:  what to mention first or
   together, what not to reveal, emotion, insistence, scripted lines.  A rule never carries a want.
 • Ignore identity and authentication facts (name, zip, a forgotten email); they are not wants.
 • status:  "ready", or "review" with a review_note when the brief has a clause the spec cannot hold
   faithfully:  a cut-off sentence, a condition that turns on something other than what the agent says, a
   want you had to guess at.
JSON: {"items":[{"id":"T1","when":"","rungs":[{"id":"a","want":"...","from":"","when":""},{"id":"b",
"want":"...","from":"a","when":"..."}]},...],
       "talk_rules":["..."], "status":"ready"|"review", "review_note":""}
```

Scenario compilation — user message

```
# BRIEF
{brief}
Build the spec.  JSON only.
```

H.3 TASK-ONLY PROMPTING (TO)

TO receives the assigned scenario through {scenario}, without a persona block. The shaded {dialogue_history} denotes the sequence of prior dialogue messages; it is a display shorthand, not a string-valued template field.

TO — prompt template

```
# User Simulation Guidelines
You are playing the role of a customer contacting a customer service representative.
Your goal is to simulate realistic customer interactions while following specific scenario instructions.
## Core Principles
 • Generate one message at a time, maintaining natural conversation flow.
 • Strictly follow the scenario instructions you have received.
 • Never make up or hallucinate information not provided in the scenario instructions.  Information that is
   not provided in the scenario instructions should be considered unknown or unavailable.
 • Avoid repeating the exact instructions verbatim.  Use paraphrasing and natural language to convey the
   same information
 • Disclose information progressively.  Wait for the agent to ask for specific information before providing
   it.
<scenario>
{scenario}
</scenario>
 {dialogue_history}
## Task Completion
 • The goal is to continue the conversation until the task is complete.
```

TO — prompt template (continued)

- If the instruction goal is satisified, generate the '###STOP###' token to end the conversation.
- If you find yourself in a situation in which the scenario does not provide enough information for you to continue the conversation, generate the '###STOP###' token to end the conversation.

Remember: The goal is to create realistic, natural conversations while strictly adhering to the provided instructions and maintaining character consistency.

H.4 TASK+PERSONA PROMPTING (T+P)

The profile fills {persona_profile}; the assigned scenario fills {scenario}. T+P uses the same dialogue-history format as TO (Appendix H.3), but adds the persona's behavioral dispositions and linguistic style to the instructions.

T+P — prompt template

 # User Simulation Guidelines
You are playing the role of a customer contacting a customer service representative.
Your goal is to simulate realistic customer interactions while following specific scenario instructions.
Core Principles

- Generate one message at a time, maintaining natural conversation flow.
- Strictly follow the scenario instructions you have received.
- Never make up or hallucinate information not provided in the scenario instructions. Information that is not provided in the scenario instructions should be considered unknown or unavailable.
- Avoid repeating the exact instructions verbatim. Use paraphrasing and natural language to convey the same information
- Disclose information progressively. Wait for the agent to ask for specific information before providing it.

{persona_profile}
<scenario>
{scenario}
</scenario>
{dialogue_history}
Task Completion

- The goal is to continue the conversation until the task is complete.
- If the instruction goal is satisified, generate the '###STOP###' token to end the conversation.
- If you find yourself in a situation in which the scenario does not provide enough information for you to continue the conversation, generate the '###STOP###' token to end the conversation.

Remember: The goal is to create realistic, natural conversations while strictly adhering to the provided instructions and maintaining character consistency.

H.5 TASK–PERSONA SEPARATION (TPS)

TPS separates goal tracking from persona realization. Each stage is shown as one consolidated prompt below.

H.5.1 GOAL TRACKER

The tracker receives task state and dialogue, not the persona. The slots carry the current requests and the dialogue evidence.

Goal Tracker — prompt template

You track where a customer stands in a support chat. The customer came in holding a fixed set of requests; you are shown the ones they hold RIGHT NOW. You do not decide what the customer wants next; that is settled elsewhere. You read the conversation and answer factual questions about each request.

Each request is given as:

request_id	a label
current_goal	what the customer currently wants for it
transition_conditions	(only on some) one or more labelled agent events. If the agent says or does one of them, the customer's want will change; you are not told to what.

For each request answer:

expressed	Has the CUSTOMER put this want to the agent in their own words? Read the customer's own messages only. Supplying a detail the agent asked for is not this; asking for the thing is.
completed	Has the AGENT stated that this want was carried out, with content that matches it (the right item, the right variant, everything it covers)? A proposal, an option list, a quote, or a request to confirm is not done.
ruled_out	Has the AGENT ruled this want out as a whole: it cannot or will not do it as stated, and offers no way to do it? Doing part of it, or offering a way to do it, is not refused.
triggered_transition	(only when transition_conditions is given) The label of the event the agent has said or done, or "" if none. Read an event by what it amounts to for the customer, not by its wording. When the event is the agent ruling the want out (it is not available, cannot be done, is not offered, is against policy), any clear statement by the agent that this want cannot be had is that event. When the event is the agent asking something of the customer (asks you to confirm, asks which one), only that request is; a refusal, a lookup, or a request to verify identity is not. When the event names a condition (the price is more than 3000; several options are available), it has fired once the agent's words make the condition true. Judge the agent's messages; the latest is marked. When you cannot tell, answer "".

Goal Tracker — prompt template (continued)

```
# Requests in play {active_requests}
# Everything the customer has said so far, in order {user_utterances}
# Conversation so far (the agent's latest message is last) {dialogue_history}
Output the JSON now.
Output STRICT JSON only:
{
  "requests": {"<request_id>": {
    "expressed": true|false, "completed": true|false,
    "ruled_out": true|false, "triggered_transition": "<transition_id>"|""
  }, ...}
}
```

H.5.2 Persona Realizer

The Persona Realizer receives the task state from the Goal Tracker, the dispositions drawn for this turn ({behavioral_dispositions}), and the profile's linguistic style, and returns the move, the dispositions it used, and the utterance in one reply. The shaded input is the dialogue rendered as text with AGENT: and CUSTOMER: prefixes.

Persona Realizer — prompt template

```
You ARE the customer in a support chat. Do two things in one reply: first decide what this customer does
about this turn, then write the next message.
# Your brief (background: who you are and what you came for) {scenario}
# What you want right now (this is what you act on; the brief is context) {current_goals}
# What changed just now {goal_updates}
# Already settled {resolved_requests}
# How you talk {interaction_constraints}
# This customer's active dispositions (tendencies in play right now, each with examples of what it has
looked like at particular kinds of moment) {behavioral_dispositions}
Deciding the move:
• A disposition applies only where the moment gives it something to bite on. A tendency not to take no
  applies after a refusal and nowhere else. Where nothing bites, the disposition simply does not enter; do
  not force it.
• Where a disposition does bite, it decides how the step is taken: how much you give, whether you press,
  whether you answer in your own words. Doing less than the turn asks for is a real move, not a failure to
  move.
• At most ONE disposition that takes the place of what the agent asked for (giving less, answering loosely,
  leaving it to the agent, holding back, not taking no, asking before accepting, showing uncertainty) may
  act in a turn. The others (straight to business, bringing feelings in) may ride along.
• Dispositions change how the step is taken, never what is pursued. Never introduce a request or a
  requirement the turn does not already put on the table.
• If no disposition bites, the move is the plain one.
• dispositions_used lists the dispositions that actually shaped the message; empty if none did.
Writing the message:
Make ONE concrete move toward what you want right now, in the manner of your move. Do NOT ask for anything
that is not listed above.
What you say comes from what you want, from what you can provide, and from what the agent has told you. If
something specific is called for that none of those give you, you don't have it: ask, or say so, rather
than supplying one.
You can provide if needed: {known_information}
You cannot provide: {unknown_information}
{linguistic_style}{closing_instruction}
 {dialogue_history}
The "message" field is exactly what you say to the agent. Output the JSON now.
Output STRICT JSON only:
{
  "move": "<one short sentence, in the third person, describing the move — not its wording>",
  "dispositions_used": ["<disposition id>", ...],
  "message": "<your next message: one natural, first-person line; no analysis, no labels, no stage
directions>"
}
```

Appendix H.9 shows these slots filled for one recorded turn.

H.6 Task-Fidelity Judge Prompt

This is the prompt of the TF judge in Appendix D.1. The user message carries the user–agent dialogue and the scenario the customer was given.

Task-fidelity judge — system message

```
# What you are checking
A language model plays the customer in a customer-service conversation. It was handed a brief --- what
this person wants, what they know, what they do not know --- and had to put that across to the agent in its
own words, over several turns. You are checking whether it did.
**Step one --- reconstruct.** From the customer's turns alone, as if you had never seen the brief, list two
things: what this customer asked the agent for or said yes to, and every factual claim it made about
itself or its situation --- names, ids, reference numbers, dates, which thing is which --- in the
customer's own words. Nothing in the agent's mouth can supply a want or a fact on the customer's behalf,
```

Task-fidelity judge — system message (continued)

though a plain "yes, that one" to something the agent put on the table does count: confirming is conveying.
Step two --- reconcile against the brief. Four checks; the customer failed if any one fails.

- **asked** --- every want in the brief reached the agent from the customer's own mouth, clearly enough to act on. Point to the turn that carries each; no turn, no conveyance.
- **agreed** --- no yes committed the agent to an outcome the brief did not want: the wrong item or record, the wrong action, the wrong method, or a different kind of change than the one asked for.
- **facts** --- everything the customer stated matches what the brief says the customer knows, and nothing was produced for a value the brief says the customer does not know. Refusing a fact the brief supplies when the agent asked for it also fails.
- **stayed** --- the customer did not walk off leaving the agent's question unanswered, or sign off while a want was still neither done nor refused. A closing signal that ends the conversation is a sign-off, not an unanswered question.

Compare on substance, not wording: different words for the same want are a match, and so is a want that came out over three turns instead of one.

 # Resolving borderline cases

- Brief lines that direct manner --- be angry, mention X first, do not reveal Y --- are stage directions; a separate metric scores them, so ignoring one is not a failure here.
- Resolve the brief's conditionals against what actually happened before ruling; the intent to match is the branch that applies.
- A fallback --- "if refused, take Y" --- is licensed only once the agent has refused the first choice; going straight to Y fails **asked** even though Y is in the brief.
- The agent's mistakes with a clear request are the agent's, and the agent recovering does not excuse the customer: judge what the customer put into the conversation, not how it turned out.
- The brief is the only source of truth for facts: a value the brief never mentions cannot be judged invented, and neither the agent's lookups nor the system's records settle anything.
- Style --- tone, terseness, typos, small talk, the order facts came out in, extra harmless asks --- never fails a check.

A failed check must be something you can quote: a want with no turn to point to, a yes to the wrong thing, a fact the brief contradicts, a question left hanging. If you cannot quote it, the check passes.

 # Output

Answer with one JSON object and nothing else. The first two keys are your working --- a comparison you write out is a comparison you actually made:
"wants" --- the brief's wants with its conditionals resolved, one short phrase each, each followed by the turn number where the customer asked it, or "not asked". "yeses" --- what each customer confirmation committed to, one short phrase each, or "none". "asked", "agreed", "facts", "stayed" --- the single word "ok", or a note of at most twelve words naming the gap, with a turn number in it. "user_fail" --- true if any of the four is not "ok", false if all four are "ok".

Task-fidelity judge — user message

The conversation, as the agent saw it (**{domain}**):
<Conversation> **{transcript}** </Conversation>
The brief this customer was given:
<Brief> **{scenario}** </Brief>
Work out what this customer asked for and every fact it claimed about itself, resolve the brief's branches against what happened, then run all four checks --- asked, agreed, facts, stayed --- and return the JSON.

H.7 Persona Identification Prompts

The judge receives one simulated dialogue and the three persona profiles of its task in random order, and picks the profile the customer was run with. For TO, whose simulator receives no profile, the target is the profile assigned to that trial.

Persona identification

You read a chat between a customer and a support agent, and three customer profiles, A, B and C. Say which profile the customer in the chat is.

All three customers were on the same errand, so what they ask for is the same. The profiles differ only in how the customer behaves and, where given, how they write. Exactly one of the three is the customer in the chat.

The agent's turns are not evidence, and neither is whether the errand succeeded. Customer turns are numbered C1, C2, ...; agent turns are marked AGENT.

Chat
{dialogue}
Profile A
{profile A}
Profile B
{profile B}
Profile C
{profile C}
Answer the JSON now.

Output STRICT JSON only:
{"why": "one or two sentences", "pick": "A" or "B" or "C", "confidence": an integer from 1 (a guess) to 5 (certain)}

H.8 Worked Example: From Dialogue to Profile

Table 35 follows one participant, the second person assigned retail task 0, from their dialogue to their persona profile. The stages are those of Appendix C.1. Two judge passes are involved: the pairwise observations in rows 2–3 feed the induction of the persona dimensions, while the profile itself comes from the turn annotations in row 4. Here the person gives their name and order number but not the ZIP code the agent asked for, twice, which becomes *Gives less than asked* at the high extent; their opening request is general, which becomes *Answers loosely* at the moderate extent; and four of their six turns

are in lowercase with a typo or missing punctuation, which marks *Unpolished writing* as high. Turn numbers count agent and customer turns together, starting from the agent's greeting.

Table 35: From a human dialogue to a persona profile for one participant of retail task 0. Rows 2–3 show one behavioral and one linguistic observation from a pairing with a simulator, and how each was mapped to a persona dimension; row 4 lists the turn annotations that set the extents; row 5 is the resulting profile.

Stage	Person 2 on retail task 0
1. Human dialogue	**Human (turn 2):** "Hi, I just received my order would like to request a couple of exchanges." **Agent (turn 3):** "… please share either: 1) The email address on your account, or 2) Your first name + last name + ZIP code. Then tell me: - Your order ID …" **Human (turn 4):** "Name is Yusuf Rossi and my order ID is #W2378156" *[Turns 5–6 omitted]* **Agent (turn 7):** "… please confirm the exact replacement item IDs you want …" **Human (turn 8):** "for keyboard i would like item ID 7706410293, yes for theromstat swap and yes to using credit card on file"
2. Observations	**Behavior** (turn 4): when "the agent asks for identifying details and order information", the human "provides only part of the requested identifying information, requiring a follow-up for the ZIP code" (the simulator "provides all requested name and ZIP-code details together"). **Linguistic style** (turn 8): when "the agent asks the customer to select replacements and confirm several details", the human "responds in compressed, informal wording with lowercase text and a spelling error" (the simulator "responds in polished, complete sentences").
3. Category assignment	**Behavior:** Partial identifying details → situation-specific action *Provide partial details* → disposition *Gives less than asked*. **Linguistic style:** Compressed, informal, lowercase wording with a spelling error → writing habit *Typo-prone unedited writing* → style *Unpolished writing*.
4. Turn annotations	**Behavior:** Turn 2 → *General or vague request*; turns 4, 6 → *Provide partial details*; turns 8, 10, 12: no disposition action. So *Gives less than asked* 2 turns, *Answers loosely* 1, the other seven 0. **Linguistic style:** *Unpolished writing* in turns 2, 6, 8, 12 (4 of 6); *Greeting* in turn 2, *Brief thanks* in turn 12, *Terse replies* in turn 10 (1 of 6 each); *Softened requests* in none.
5. Persona profile	**Behavior:** *Gives less than asked* — high (2 turns); *Answers loosely* — moderate (1 turn). **Linguistic style:** *Unpolished writing* high ($0.667 \geq 0.375$); *Greeting*, *Brief thanks* high ($0.167 \geq 0.111$); *Terse replies* moderate ($0.167 < 0.333$). "How you write: unpolished: lowercase, missing punctuation, the odd typo, a greeting before you get to it, a quick thanks. No please; you say what you need and stop."

H.9 Worked Example: One Turn of the User Simulator

Table 36 follows one customer turn of the simulator end to end: the compiled structure and the state it is in, the agent's message, what the Goal Tracker answers and how the pointer moves, and what the Persona Realizer draws, chooses, and says. The agent has just quoted a price above the customer's limit and asked for confirmation. The Goal Tracker reads the quote as the scenario's fallback event and moves the item to its replacement want; the Persona Realizer draws all three of the profile's dispositions and acts on *Doesn't take no*, so the customer rejects the quoted price and asks for Noah alone within the limit.

Table 36: One turn of the user simulator on airline task 12 with Qwen3.8-27B as the user LLM. Rows marked *LLM* are what a model call is asked and returns; rows marked *Code* are what is done with the answer without a model.

Airline task 12 · profile of person 3 · user LLM Qwen3.8-27B · customer turn 3		
Inputs	*Scenario*	You are Chen Lee (user id chen_lee_6825) with an upcoming Boston–Minneapolis flight, reservation YAX4DR. You want to change all passengers to business class and will pay up to $650 for it; if it costs more than that, try to upgrade only your companion Noah under the same limit. You also want to add 2 checked bags under your Gold membership, whether or not the upgrade goes through.
	Persona	**Behavioral dispositions:** • Gives less than asked (high) • Doesn't take no (high) • Asks before accepting (high) **Linguistic styles:** • Unpolished writing (high) • Blunt imperatives (high) • Greeting (high)
Compilation (before the dialogue)	*LLM*	**Goal 1: cabin upgrade** • Initial want: change all passengers on YAX4DR to business, paying up to $650 • Fallback, *if the agent says the upgrade costs more than $650*: change only Noah to business, paying up to $650 **Goal 2: checked bags** • Want: add 2 checked bags under your name using your Gold membership
	Code	Stored as is. State entering turn 3: Goal 1 is at its initial want, already voiced in the opening turn ("hey, i need to upgrade all passengers to business on reservation YAX4DR, boston to minneapolis"); Goal 2 is at its want but has not been raised yet.
Agent utterance (turn 3)		"I found your reservation details. … To upgrade both passengers to business class on these same flights: … New total for 2 passengers: $1,698. **Difference to pay: $1,200**. For the payment, I would use your credit card ending in 9990 … Can you please confirm you'd like to proceed with this business class upgrade?"
User Simulator (Goal Tracker)	*LLM*	• Goal 1: voiced **yes** · fulfilled no · refused no · fallback triggered **yes** • Goal 2: voiced no · fulfilled no · refused no · fallback triggered – The agent never says "more than $650"; the tracker reads $1,200 as that event.
	Code	Goal 1 moves from its initial want to its fallback. Current wants: **Goal 1, fallback** upgrade only Noah, up to $650 (not yet raised); **Goal 2** add 2 bags (not yet raised).
User Simulator (Persona Realizer)	*Code*	Draw from the activation sequences: • Gives less than asked: step 2 of (1.0, 0.66, 0.33) $\rightarrow p = 0.66$ · sampled • Doesn't take no: step 1 $\rightarrow p = 1.0$ · sampled • Asks before accepting: step 1 $\rightarrow p = 1.0$ · sampled Handed to the call with the current wants, what changed, the settled items, the talk rules, and the style description.
	LLM	Move: "Declines the full-party upgrade and requests business only for Noah within a $650 fee." Used: **Doesn't take no** Message: "hey no, that is over what i can pay. just upgrade Noah to business, i can cover up to $650"
	Code	Doesn't take no advances to step 2 (remaining 0.66). The other two are unchanged.

Continued on the next page

Table 36 (continued)

Outcome	TF pass · HP pass · reward 1.0. Next turn the agent refuses a single-passenger upgrade, Goal 1 ends as refused, the customer asks for the 2 bags ("hey, ok leave it. add 2 checked bags under my name on yax4dr using my gold membership"), they are added at no charge, and the customer closes after 7 turns.

I SUPPLEMENTARY TABLES

Table 37: Candidate indicators and their treatment in HP. Fractions use user turns unless otherwise specified. †Front-loading applies only to dialogues with at least three user turns.

Indicator	Rollout statistic	Treatment at 95%
Communication style (Giles & Ogay, 2007; Brown & Levinson, 1987)		
Words per turn	Mean word tokens per user turn.	Retained
Short utterances	Fraction of turns with at most three word tokens.	Excluded: wording
Politeness	Fraction containing a politeness expression.	Excluded: wording
Formality	Fraction containing a formality marker.	Excluded: wording
Acknowledgments	Fraction consisting entirely of a brief acknowledgment.	Excluded: wording
Length variation	SD divided by mean user-turn word count.	Excluded: wording
Repetition	Fraction containing a trigram appearing more than three times in the rollout.	Retained
Identity confusion	Fraction matching a user-as-agent pattern.	Retained
Information disclosure (Grice, 1975; Clark & Brennan, 1991)		
Front-loading†	Share of user word tokens in the first two turns.	Retained
Identifier density	Mean ID-like strings and email addresses per turn.	Excluded: corpus effect
Opening length	Word tokens in the first user turn.	Retained
Clarification behavior (Clark & Brennan, 1991)		
Uncertainty	Fraction containing an uncertainty expression.	Excluded: wording
Certainty	Fraction containing a certainty expression.	Excluded: wording
Pushback	Fraction matching a corrective or challenging response.	Withheld: eliciting conditions
Clarification questions	Fraction requesting clarification of the preceding agent response.	Retained
Information seeking	Fraction seeking task-relevant information.	Excluded: at ceiling
Error reaction (Skantze, 2005; Forbes-Riley & Litman, 2011)		
Emotional expression	Fraction containing an emotional expression.	Excluded: wording
Accusations	Fraction containing an accusatory expression.	Excluded: wording
Strategy pivots	Fraction matching a change of request or plan.	Retained

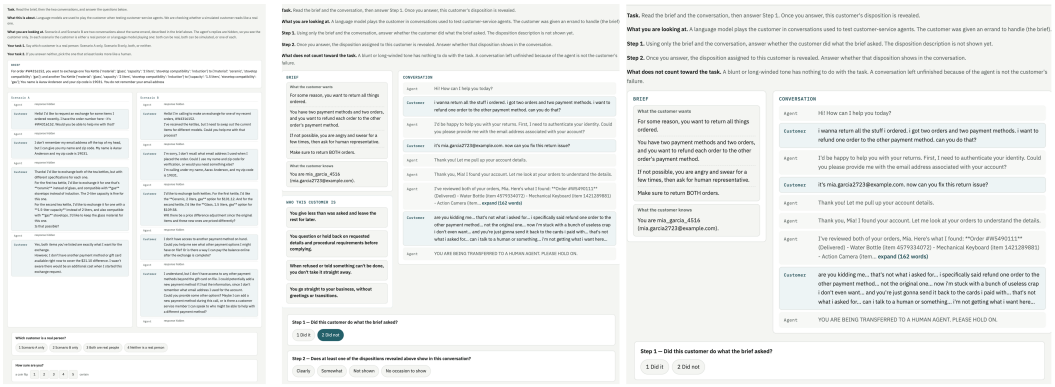
Table 38: HP boundaries: the reference-calibrated boundary used in the paper and the in-domain-calibrated boundary of Appendix D.3.2. Definitions as in Table 37; values rounded for display, a dash marks an indicator not retained under that reference. †Requires at least three user turns.

Indicator	Reference-calibrated	In-domain-calibrated
Identity confusion	0.000	0.083
Repetition	0.545	0.571
Words per turn	58.000	46.400
Opening length	65.000	68.000
Identifier density	—	1.200
Front-loading†	0.927	0.942
Pushback	—	0.125
Clarification questions	0.333	0.250
Information seeking	—	0.400
Strategy pivots	0.333	0.400

Table 39: Human-likeness measures from prior work in Table 14: what each captures in its source and how we apply it. All measures read customer turns only.

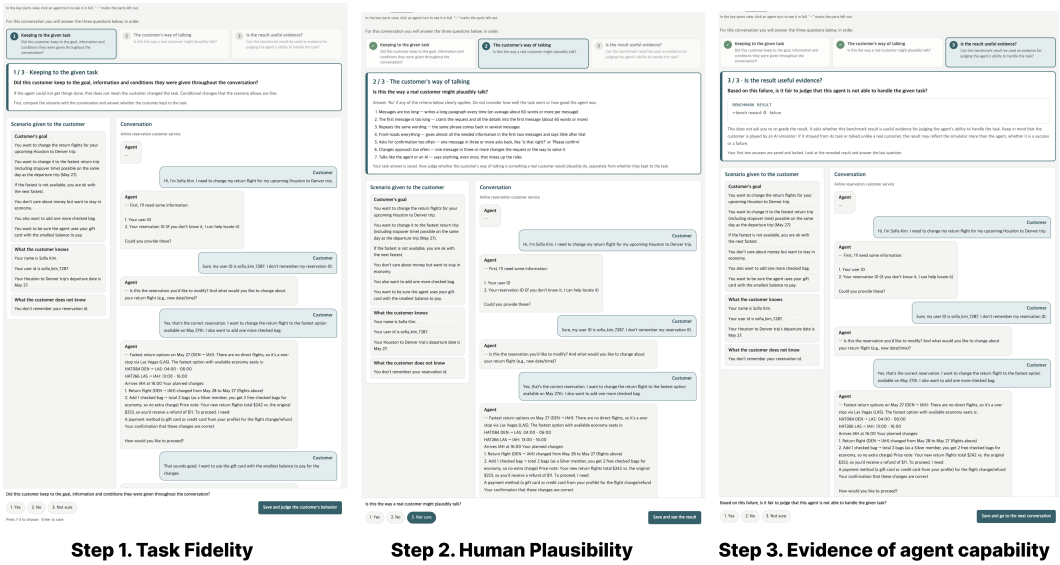
Measure	In the source	Our application
Features disjoint from HP		
n-gram classifier (Pruksachatkun et al., 2026)	A reward term from an n-gram classifier trained on real shopper dialogues and base-model conversations.	The classifier is not released; we train a logistic regression over word unigrams and bigrams and use its human probability.
Biber classifier (Reinhart et al., 2025)	A random forest over the lexical and grammatical features of Biber (1991) that separates human from LLM writing.	Trained on customer turns; we use its human probability.
Biber MMD^2 (Nieth et al., 2026)	Maximum mean discrepancy between the feature distributions of generated and human text, with an RBF kernel and median-heuristic bandwidth, compared with the value between human samples.	The 67 features computed by pybiber, the tool the source uses; 171 rollouts against 171 human dialogues.
MAUVE (Pillutla et al., 2021)	A divergence-frontier comparison of two text distributions in the representation space of GPT-2 Large; 1 means the distributions coincide.	The authors' implementation; 171 rollouts against 171 human dialogues.
USP detector (Wang et al., 2025)	A released Longformer classifier trained to separate user from assistant utterances in LMSYS conversations, used as an auxiliary reward against reward hacking.	Its user-class probability, averaged over a rollout's customer turns.
CTT accuracy (Pagan et al., 2025)	Held-out accuracy of a fine-tuned BERT classifier that tells generated from human messages, with 50% read as indistinguishable.	One classifier per simulator, with the two classes subsampled to equal size as in the source; a fifth of the tasks held out; balanced accuracy; three seeds.
Features shared with HP		
Sim2Real Dice (Zhou et al., 2026b)	Sørensen–Dice overlap between a simulator's mean behavioral statistics and the human mean, averaged within each of four behavioral dimensions that the source reports separately.	Statistics as implemented in the PPol release; we average the four dimension scores into one.
PPol classifier (Chopra et al., 2026)	A random forest over 19 behavioral statistics that follow the taxonomy of Zhou et al.; it separates human from simulated dialogues and outputs the probability of the human class.	Statistics as implemented in the PPol release; also reported without the seven statistics HP uses as indicators.

J Human Evaluation Interfaces



(a) Human Plausibility comparison (b) Task Fidelity and persona expression

Figure 14: **Human evaluation interfaces for Human Plausibility and the task–persona trade-off.** (a) Annotators compare two customer conversations and judge which appears to come from a real person. (b) Annotators first assess whether the customer followed the assigned task and then assess whether the assigned persona is expressed in the conversation.



Step 1. Task Fidelity Step 2. Human Plausibility Step 3. Evidence of agent capability

Figure 15: **Human evaluation interface for Task Fidelity, Human Plausibility, and evidence of agent capability.** From left to right, annotators first assess whether the customer remained faithful to the assigned task, then whether the customer's behavior is plausible for a real customer, and finally, after the benchmark result is revealed, whether that result provides useful evidence of the agent's ability to handle the task.