DIAL: Direct Iterative Adversarial Learning for Realistic Multi-Turn Dialogue Simulation

Anonymous ACL submission

Abstract

Realistic user simulation is crucial for training and evaluating multi-turn dialogue systems, yet creating simulators that accurately replicate human behavior remains a significant challenge. An effective simulator must expose the failure modes of the systems under evaluation. This work introduces Direct Iterative Adversarial Learning (DIAL), an adversarial framework that iteratively enhances user simulator realism through a competitive dynamic between a generator (user simulator) and a discriminator. When applied to mental health support, a domain characterized by diverse failure types and a critical dependence on realistic user behavior for failure detection, DIAL restores lexical diversity diminished by supervised fine-tuning and drastically reduces discriminator accuracy. The resulting simulator exhibits a strong correlation between simulated and real failure occurrence rates while maintaining low distributional divergence of failure modes. These findings indicate that DIAL is a promising method for developing realistic user simulators in multi-turn dialogue, facilitating reliable and cost-effective system evaluation prior to deployment.

1 Introduction

Task-oriented dialogue (TOD) systems facilitate multi-turn conversations to help users achieve specific objectives (Schatzmann et al., 2006). The emergence of large language models (LLMs) has highlighted the need for robust evaluation methodologies that extend beyond single-turn metrics, as task success depends on maintaining context across extended dialogue. Traditional evaluation methods, such as live testing, are costly, time-intensive, and may expose users to suboptimal system behavior.

User simulation provides a viable alternative, as synthetic agents that mimic human conversational behavior enable rapid iteration and systematic exploration of edge cases (Verma et al., 2022; Yu et al., 2023). However, developing simulators that accurately capture the nuanced, diverse, and goal-oriented nature of human conversation remains challenging. Simulators that do not reflect realistic user behavior fail to reveal system failure modes during development, resulting in policies that perform well in simulation but poorly with real users.

This work addresses the challenge of developing realistic user simulators through Direct Iterative Adversarial Learning (DIAL). Drawing inspiration from Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), DIAL trains a discriminator to differentiate between simulated and real user sessions, and uses discriminator feedback to refine the simulator via Direct Preference Optimization (DPO) (Rafailov et al., 2023). Unlike prior adversarial dialogue approaches that rely on policy gradient methods (Li et al., 2017; Zhao et al., 2021), DIAL avoids training instability and naturally prevents reward hacking through its discrete iteration structure (see Section 3 for details).

DIAL is primarily evaluated using our mental health support chatbot. The mental health domain contains a diverse range of failure types, which require realistic user simulation for effective detection of system shortcomings. Our results reveal several key findings:

- **DIAL methodology**: Three iterations of DIAL refinement progressively restore lexical diversity diminished by supervised learning, enhance distributional similarity, and reduce discriminator accuracy from near-perfect to near-random levels.

- **Failure-sensitive evaluation**: Domain-specific fine-tuning, in conjunction with DIAL refinement, is critical for capturing realistic user behaviors, including negative and resistant patterns that zero-shot models do not exhibit. This enables accurate identification of diverse chatbot failure modes.

- **Strong predictive validity**: The optimal DIAL-trained simulator demonstrates a strong correlation with real model performance, facilitating reliable offline evaluation.

These findings indicate that DIAL is a promising methodology for developing realistic user simulators in multi-turn dialogues, with potential applications in RL-based policy optimization and reliable offline evaluation.

2 Related Work

2.1 Multi-Turn Dialogue Optimization

Although RLHF (Ouyang et al., 2022) has shown that language models can be aligned with human preferences, its optimization is generally limited to single-turn responses. DPO provides an alternative by directly optimizing the policy without explicit reward modeling, thereby simplifying training while preserving effectiveness.

Enhancing dialogue systems for sustained conversational quality necessitates approaches that extend beyond single-turn reward modeling. Recent methods encompass training on imagined conversations and hindsight regeneration (Hong et al., 2023, 2024), token-level policy optimization (Shani et al., 2024), hierarchical decomposition of planning and execution (Zhou et al., 2024), and offline reinforcement learning for language generation (Snell et al., 2023). The effectiveness of these approaches is highly contingent on the quality of the training environment, with realistic user simulators being indispensable for both policy optimization and evaluation (Nam et al., 2025).

2.2 User Simulation for Task-Oriented Dialogue

User simulation has played a central role in TOD research since the development of early agenda-based systems, which modeled behavior through goal-directed dialogue acts (Schatzmann et al., 2006, 2009). Subsequent neural approaches enabled more flexible simulation by capturing dialogue history without rigid structural constraints (El Asri et al., 2016; Kreyssig et al., 2018), and demonstrated that policies trained on neural simulators can achieve higher success rates with real users. Recent benchmarking in mental health support (Pombal et al., 2025) reveals that LLMs encounter difficulties with extended interactions and patients exhibiting severe symptoms, highlighting the necessity for realistic simulators that reflect the full complexity of human emotions and behaviors.

LLM-Based Simulation: LLMs have significantly advanced user simulation by leveraging in-context learning and prompting methods (Li et al., 2022; Davidson et al., 2024), achieving goal success rates comparable to human-bot interactions while offering greater linguistic diversity and requiring minimal seed dialogues. Nevertheless, prompting-only strategies are prone to hallucinations and goal drift. Fine-tuning mitigates these issues through techniques such as reducing behavioral inconsistencies, jointly optimizing policy and natural language generation (NLG) for domain transfer, and employing dual-model architectures with chain-of-thought reasoning (Sekulić et al., 2024; Lin et al., 2022; Luo et al., 2024).

RL-Based Simulator Training: While supervised learning dominates simulator training, reinforcement learning methods optimize for task-level objectives beyond mere imitation. Multi-agent reinforcement learning facilitates joint system-user training (Takanobu et al., 2020; Liu and Lane, 2017), while adversarial training introduces competitive dynamics between simulators and discriminators (Keizer et al., 2023; Li et al., 2017; Zhao et al., 2021). Li et al. (2017) employ REINFORCE with Monte Carlo search to backpropagate discriminator rewards, but report that this leads to unstable training and necessitates teacher forcing as a regularizer. Zhao et al. (2021) also utilize GAN-based policy gradient methods for user simulation in recommendation systems. The present work contributes methodologically by extending the adversarial paradigm to LLM-based simulators through DIAL, with a key distinction in optimization: DPO is applied to discriminator-derived preference pairs instead of policy gradient methods. DIAL circumvents the instability associated with policy gradients in text generation and inherently prevents reward hacking (see Section 3).

Evaluation Methodology: Traditional TOD evaluation, including recent end-to-end frameworks such as AutoTOD (Xu et al., 2024) and ProTOD (Dong et al., 2025), relies on task-level metrics such as success rate and slot accuracy measured on standard benchmarks like MultiWOZ (Budzianowski et al., 2018), SGD (Rastogi et al., 2020), and tau-bench (Yao et al., 2024). These metrics effectively capture transactional outcomes but provide limited insight into conversational quality. Our work addresses the complemen-

tary problem of building realistic user simulators to evaluate such systems, focusing on failure mode detection rather than transactional success, and advances this evaluation paradigm through three key contributions: (1) explicit modeling of conversational feature occurrence rate distributions to measure alignment on fine-grained failure modes, (2) validation of predictive utility through correlation analysis, and (3) discriminator accuracy as a direct measure of distributional realism.

3 Methodology

3.1 Problem Formulation

We formalize user simulation as a conditional generation task. Given conversation history $h_t = \{(u_1, a_1), ..., (u_{t-1}, a_{t-1}), a_t\}$ consisting of previous user utterances u_i and agent responses a_i, plus user context c (e.g., interaction history and background for the user), the simulator generates the next user utterance $u_t \sim \pi_\theta(u_t|h_t, c)$.

We train π_θ to generate simulated conversations indistinguishable from real user conversations. To this end, we bootstrap an initial simulator from real conversations, construct prompts with appropriate context, and iteratively alternate between training discriminators to detect simulated conversations and refining simulators to evade detection.

3.2 Context Construction for User Simulation

A critical design choice for realistic user simulation is the construction of context c that conditions the simulator's behavior. Effective context should provide: (1) role and behavioral guidelines, (2) user background and goals, (3) conversational history to maintain consistency, and (4) task-specific grounding to anchor the simulation. The specific context structure depends on the domain—transactional TOD systems might include task schemas and slot values, while open-ended domains might emphasize user profiles and session summaries. Our mental health support chatbot (detailed in Appendix A) uses a hierarchical structure spanning user profiles, multi-session summaries, and current session grounding. As we demonstrate in Section 5, rich context grounding enables fine-tuned simulators to cluster tightly with their corresponding real sessions in embedding space, while zero-shot models exhibit systematic behavioral drift.

3.3 Base Simulator Training

We begin by training a base user simulator through supervised fine-tuning (SFT) on anonymized user conversations. Starting from Llama-3.3-70B-Instruct (Grattafiori et al., 2024) as our pre-trained foundation model, we fine-tune on a dataset $\mathcal{D} = \{(h_i, c_i, u_i)\}_{i=1}^{N}$ of real user responses paired with conversation history and context:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(h,c,u)\sim\mathcal{D}}[\log \pi_\theta(u|h, c)] \quad (1)$$

3.4 Discriminator Training

To measure simulator realism, we train a discriminator D_ϕ to distinguish between simulated and real user sessions. We implement the discriminator as a token classification model which processes the full conversation and predicts after each user message whether the session is real or simulated:

$$p_t = D_\phi(\text{real}|h_t, c) \quad (2)$$

We use causal attention in the discriminator, meaning predictions are made based on the current message and all previous messages in the session. The discriminator's effectiveness stems from detecting *distributional* patterns across the conversation: repeated phrases, unnatural topic progressions, or behavioral patterns that diverge from real user dynamics.

We train D_ϕ on a balanced dataset of real and simulated sessions using binary cross-entropy loss:

$$\mathcal{L}_{\text{disc}} = -\mathbb{E}_{(h,c,y)}[y \log p + (1-y) \log(1-p)] \quad (3)$$

where $y = 1$ for real sessions and $y = 0$ for simulated sessions. A high discriminator accuracy indicates the simulator has not achieved realistic behavior; our goal is to train simulators that reduce discriminator accuracy toward random guessing.

In DIAL, each new discriminator is trained exclusively on data from the most recent simulator version. This ensures the preference pairs we generate for DPO reflect the current policy's behavior rather than artifacts from earlier iterations. Research on preference optimization supports this on-policy approach: on-policy preference pairs exhibit stronger contrastiveness and clearer preference signals (Zhao and Li, 2025), online methods require only "partial coverage" compared to the strict "global coverage" needed by offline methods (Song et al., 2024), and on-policy sampling

Algorithm 1 DIAL

Initialize user simulator π_θ from SFT
Initialize discriminator D_ϕ
for iteration $k = 1, 2, \ldots$ **do**
  Generate simulated sessions $\mathcal{S}_k$ using π_θ
  Train discriminator D_ϕ on real vs. simulated sessions from π_θ
  Compute rewards r_t and identify the N lowest-reward messages in $\mathcal{S}_k$
  Sample alternative responses at critical points to create preference dataset $\mathcal{D}_{\text{pref}}$
  Retain only those pairs satisfying $r_{\text{chosen}} > 0$ and $r_{\text{rejected}} < 0$
  Update simulator using DPO: $\pi_\theta \leftarrow \text{DPO}(\pi_\theta, \mathcal{D}_{\text{pref}})$
end for

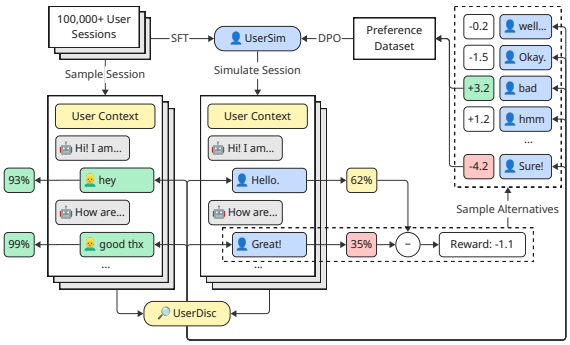

Figure 1: Overview of the DIAL training process. The simulator generates sessions that are evaluated by the discriminator to distinguish real from simulated conversations. Discriminator feedback guides preference dataset generation, which is then used to refine the simulator through DPO, creating an iterative competitive dynamic that improves realism.

efficiently concentrates probability mass on high-reward responses (Tajwar et al., 2024).

3.5 DIAL: Direct Iterative Adversarial Learning

The DIAL training process follows an iterative refinement loop, illustrated in Figure 1. Algorithm 1 provides the detailed procedure.

Optimization Method: DIAL uses DPO rather than RLHF methods (Schulman et al., 2017; Shao et al., 2024) because discriminator rewards are inherently policy-dependent and vulnerable to reward hacking. With RLHF, the simulator would continuously optimize against a fixed reward model, eventually exploiting patterns that score highly but don't reflect genuine realism. Preventing this would require retraining the discriminator throughout optimization—a substantial computational overhead. DPO avoids this problem: we generate preference pairs once per iteration using the current discriminator, then optimize against this fixed dataset. Preferences grounded in comparisons between specific alternatives are more robust than absolute reward values, and the discrete iteration structure naturally prevents reward hacking.

Reward Signal: For each user message in a simulated session, we compute a reward based on how the message changes the discriminator's prediction:

$$r_t = \log\left(\frac{p_t}{1 - p_t}\right) - \log\left(\frac{p_{t-1}}{1 - p_{t-1}}\right) \quad (4)$$

We use log-odds rather than raw probabilities to reflect the exponentially increasing difficulty of fooling the discriminator at higher confidence levels. Higher rewards indicate messages that make the session appear more realistic.

Preference Dataset Generation: For 2 messages with the lowest rewards, we sample 8 alternative responses. These alternatives are ranked by their discriminator-based rewards to create preference pairs $(u_{\text{chosen}}, u_{\text{rejected}})$ where the chosen response has the highest reward.

Dataset Filtering: After generating preference pairs, we filter to retain only samples where $r_{\text{chosen}} > 0$ and $r_{\text{rejected}} < 0$. This selects high-confidence pairs with maximum learning signal while preventing over-correction.

DPO Update: We train the new simulator using DPO, which directly optimizes the policy to prefer higher-reward responses. We use the previous iteration's simulator as the reference policy π_{ref}. Defining the implicit reward $r_\theta(h, c, u) = \log \frac{\pi_\theta(u|h,c)}{\pi_{\text{ref}}(u|h,c)}$, the DPO loss is:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(h,c,u_c,u_r)\sim\mathcal{D}_{\text{pref}}}\Big[\log\sigma\Big(\beta\big(r_\theta(h, c, u_c) - r_\theta(h, c, u_r)\big)\Big)\Big] \quad (5)$$

where u_c and u_r are chosen and rejected responses, β is the temperature parameter, and σ is the sigmoid function.

4 Experimental Setup

4.1 Domain and Data

DIAL is evaluated within the domain of mental health support, in which users pursue mental health

goals such as emotional regulation, insight development, and behavioral change through extended multi-turn conversations. This domain was selected to specifically assess simulators' capacity to reveal failure modes of TOD systems, as mental health conversations present a particularly diverse array of failure types (see Appendix F). In contrast to traditional TOD benchmarks such as MultiWOZ (Budzianowski et al., 2018) and SGD (Rastogi et al., 2020), where failure modes are primarily transactional (e.g., incorrect slot filling or task incompletion) and less reliant on realistic user simulation, the mental health support domain features complex emotional dynamics and naturalistic failure modes that require faithful user behavior for effective detection. Furthermore, production-scale data in this domain captures rare but critical phenomena, such as crisis expressions and resistance to engagement, thereby providing a rigorous testbed for evaluating simulator realism.

Our dataset consists of anonymized real user conversations in English with our mental health support chatbot, including session-level context. We additionally validate our evaluation framework on MultiWOZ 2.1 (Section 5.3), an open-source transactional TOD benchmark, to demonstrate broader applicability of the diversity and distributional metrics used in this work.

4.2 Model Configurations

We train simulators by fine-tuning Llama-3.3-70B-Instruct using LoRA (Hu et al., 2022) on datasets of varying sizes (10K, 40K, 160K sessions) to explore data scale effects. We apply DIAL on the final checkpoint, producing UserSim-160K-DIAL-it1, UserSim-160K-DIAL-it2, and UserSim-160K-DIAL-it3 through successive refinement iterations. For comparison, we evaluate five zero-shot base models: GPT-4o (Hurst et al., 2024), GPT-5.2-Chat (Singh et al., 2025), Kimi-K2-Instruct (Bai et al., 2025), Qwen-2.5-7B-Instruct (Yang et al., 2025), and Llama-3.3-70B-Instruct. Training hyperparameters are provided in Appendix C.2.

The discriminators (UserDisc-SFT, UserDisc-DIAL-it1, UserDisc-DIAL-it2, UserDisc-DIAL-it3) are token classification models trained from the Llama-3.1-8B-Instruct backbone.

For each configuration, we generate 2,000 simulated sessions: 1,000 unique session contexts drawn from real user profiles and conversation histories × 2 simulations per context, with early stopping when conversations naturally conclude. To isolate simulator quality effects, we use a fixed set of 14 chatbot models to simulate conversations.

4.3 Evaluation Metrics

4.3.1 Linguistic Features

For all linguistic features, we calculate metrics for each session individually and report distributional statistics (mean, median, standard deviation) across sessions to capture both typical behavior and variability.

Word Entropy (H): Vocabulary diversity using Shannon entropy: $H = -\sum_w p(w) \log p(w)$, where $p(w)$ is the probability of word w in the session's vocabulary. Higher entropy indicates more diverse word usage.

Type-Token Ratio (TTR): The ratio of unique words (types) to total words (tokens) in a session, measuring lexical diversity. Higher TTR indicates greater vocabulary variety.

N-Gram Diversity (D-2, D-3): The ratio of unique n-grams to total n-grams in a session. We report bigram diversity (D-2) and trigram diversity (D-3), measuring phrase-level variety and reducing sensitivity to repetitive phrasing.

MAUVE: Principled distributional metric for mode collapse detection (Pillutla et al., 2021). It embeds utterances, quantizes them into clusters, and computes the area under the divergence frontier between simulated and real distributions. Higher values indicate closer alignment. Configuration details are provided in Appendix B.2.

4.3.2 Behavioral Metrics

Various behavioral metrics were adopted to demonstrate simulator realism:

User Action Distribution: Conversations are annotated with task-relevant user actions using GPT-5-nano to classify each individual user message according to our action taxonomy (see Appendix E). Since individual messages can have multiple actions, the action distributions are not normalized.

Issue Rate Analysis: A critical requirement for realistic user simulation is the ability to expose chatbot failure modes. We evaluate simulators by measuring how well they replicate the rate and distribution of problematic conversational features ("issues") that chatbots exhibit with real users. Issues are identified using GPT-5 with low reasoning effort. The analyzer identifies issues from a detailed taxonomy and highlights the specific messages where each issue occurred, enabling proper

credit assignment (see Appendix F). We compare issue rates and distribution between simulated and real conversations to validate that our simulator accurately reflects the chatbot's real-world performance and can reliably identify system weaknesses before deployment. We use KL divergence to measure distributional alignment.

To validate the reliability of the issue analyzer, we conducted a human evaluation on 100 randomly sampled sessions, measuring annotator agreement with detected issues and the rate of omitted issues across three frontier LLMs (Appendix D). GPT-5 achieved the lowest omission rate with the second-highest agreement, supporting its use as our primary annotation model where comprehensive issue detection is prioritized. Furthermore, systematic biases inherent in single-judge LLM evaluation can be efficiently mitigated through round-robin judge assignment (Zhu et al., 2026), which eliminates judge bias while matching the cost of single-judge evaluation.

Similar to the discriminator, the issue analyzer examines whole sessions rather than isolated messages. This enables numerous evaluation dimensions that would be impossible with isolated message analysis: detecting circular conversations, identifying premature topic changes, and assessing whether mental health progress occurs naturally over the conversation arc.

4.3.3 Discriminator Performance

We measure discriminator accuracy using discriminators trained on sessions from their corresponding simulators. Each discriminator is trained on 1,600 sessions with the same hyperparameters and evaluated on 400 sessions.

4.3.4 Human Judgment

We complement discriminator metrics with a human evaluation of realism, drawing inspiration from the Turing test. For each simulator, we sample 100 pairs of simulated and real sessions, presented side by side in random order. Annotators are asked to identify which session is real. The success rate is the proportion of correct identifications. Human judgment exhibits high variance and depends on familiarity with simulator-specific quirks; we therefore use a fixed panel of in-team annotators with prior experience using the simulators.

Table 1: Linguistic diversity features comparing real users with simulators. Best fine-tuned values are in **bold**, second best are underlined.

Simulator	H	TTR	D-2	D-3	MAUVE
Human	6.53	.499	.872	.965	–
Zero-shot					
GPT-4o	7.12	.408	.837	.956	.128
GPT-5.2	7.73	.266	.738	.922	.015
Kimi-K2	7.65	.416	.851	.965	.118
Qwen-2.5-7B	7.07	.390	.791	.914	.158
Llama-3.3-70B	7.02	.268	.651	.831	.121
UserSim					
10K-SFT	6.14	.406	.756	.888	.936
40K-SFT	6.14	.430	.782	.907	.919
160K-SFT	6.18	.382	.728	.867	.944
160K-DIAL-it1	6.38	**.471**	.839	.944	.963
160K-DIAL-it2	**6.52**	.439	.821	.937	**.984**
160K-DIAL-it3	6.51	.461	**.841**	**.948**	.974

5 Results

5.1 Fine-tuned vs. Zero-shot Comparison

As a foundation for DIAL, we first establish that domain-specific SFT is necessary for realistic simulation. The embeddings of conversations simulated by base models form distinct clusters separate from their real counterparts (Appendix B.1). The fine-tuned simulators' outputs, on the other hand, form tight clusters defined by shared context, demonstrating effective grounding. Zero-shot models produce unrealistically compliant users, as evidenced by significantly lower issue rates (Figure 2) and over-representation of positive mental health indicators (Appendix B.4).

As shown in Table 1, which compares linguistic features across simulators, SFT exhibits non-monotonic effects on lexical diversity. N-gram diversity (D-2, D-3) and TTR improve from 10K to 40K samples as the model learns to use conversational context to improve diversity. However, mode collapse emerges from 40K to 160K samples as continued training learns common patterns at the expense of rare variations. MAUVE is substantially higher for fine-tuned simulators than for zero-shot models, indicating much better distributional alignment with real user utterances. For behavioral alignment, fine-tuned simulators dramatically outperform zero-shot base models (see Appendix B.4). To evaluate how well simulators surface chatbot issues, we measure issue rates as the percentage of chatbot messages containing issues. Figure 2 shows that fine-tuned simulators produce rates closely matching real conversations,

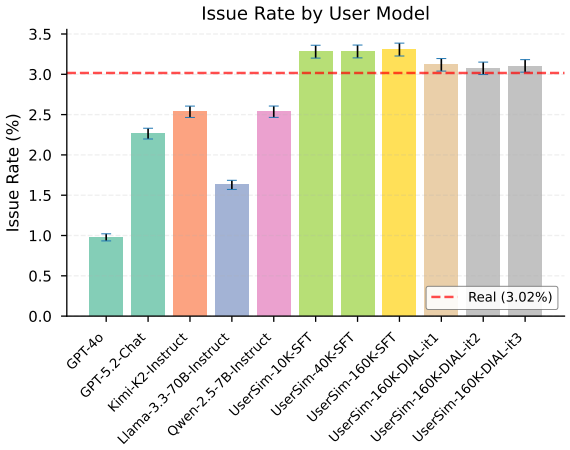

Figure 2: Issue rate comparison across simulated and real sessions. Error bars represent standard error.

while zero-shot models show substantially lower rates. Detailed issue category discrepancies are provided in Appendix B.3.

5.2 DIAL Training Improvements

Three iterations of DIAL produced progressive improvements:

Linguistic Diversity: DIAL successfully recovers lexical diversity while maintaining behavioral alignment (see Appendix B.4). However, non-monotonic improvement across iterations indicates the need for careful hyperparameter tuning. We employ conservative dataset filtering to mitigate over-correction and maintain stability.

Distinguishability: DIAL dramatically reduced discriminator accuracy and the success rate of human judgment (Table 2). This confirms that DIAL iterations remove systematic artifacts and produce strong distributional alignment with real behaviors throughout sessions.

Predictive Validity: To evaluate how well simulators enable reliable offline evaluation, we measure correlation between simulated and real issue rates across diverse chatbot models, and KL divergence between simulated and real issue category distributions. Table 3 shows that DIAL produces substantial improvements in both metrics. UserSim-160K-DIAL-it2 achieves the strongest correlation, while UserSim-160K-DIAL-it3 achieves the lowest distributional divergence. This provides compelling evidence that DIAL-trained simulators are significantly more realistic and effective at identifying chatbot problems in offline testing compared to their base fine-tuned counterparts. Notably, aggregating predictions across all simulator models achieves both a strong correlation and low distributional divergence, suggesting that ensembling multiple simulators could provide even more robust evaluation. Beyond correlation, we validate predictive utility by replicating a recent model A/B test (Appendix B.6).

5.3 TOD Benchmark Validation

To validate generalizability beyond our primary domain, we evaluate baseline user simulation approaches on MultiWOZ 2.1. Using ConvLab-3 (Zhu et al., 2023), we compare five baselines on 1,000 test dialogues: (1) **Rule-based** agenda-based simulation (Schatzmann et al., 2009) with a rule-based system; (2) **TUS** (Transformer User Simulator) (Lin et al., 2021) with a rule-based system; (3) **GenTUS** (Generative TUS) (Lin et al., 2022) with a rule-based system; (4) **Zero-shot** (Llama-3.1-70B-Instruct) for both user simulation and response generation; and (5) **SFT** (Llama-3.1-70B-Instruct fine-tuned on 8,438 train dialogues) for both user simulation and response generation.

Table 4 presents diversity, distributional, and task success results. We additionally apply DIAL on the SFT user simulator, with both SFT and DIAL models evaluated against the SFT response generator. DIAL-it3 achieves the best MAUVE score, substantially improving distributional alignment over SFT while also recovering lexical diversity. Notably, SFT is limited to a single epoch, as further training induces severe mode collapse where the simulator degenerates into repeating identical sentences; DIAL avoids this failure mode while continuing to improve realism. For LLM-based systems, ConvLab-3 equates success with conversation completion as determined by the simulator, without verifying system responses against the database. DIAL success rates are nonetheless lower than SFT, potentially reflecting more calibrated user behavior, as less realistic simulators tend toward excessive compliance that inflates success metrics. A key difference from our primary domain is MultiWOZ's Wizard-of-Oz data collection methodology (Budzianowski et al., 2018), where crowd workers on both sides followed predefined task schemas, producing less natural and diverse interactions than genuine user conversations.

6 Discussion

As context for the substantive contributions, note that zero-shot models show severe distributional

Table 2: Discriminator performance at different conversation turns, and success rate with human judgment on the same conversations. All values are in percentage (%). MCC (Matthews Correlation Coefficient) is a measure of the correlation between the observed and predicted classifications. Best results are in **bold**, second best are underlined.

	Turn 1			Turn 5			Turn 10			Turn 25			Overall			Human
Discriminator	Acc	F1	MCC	Acc	F1	MCC	Acc	F1	MCC	Acc	F1	MCC	Acc	F1	MCC	(%)
SFT	62.0	80.1	24.3	82.3	81.6	64.7	90.2	88.6	80.1	96.1	93.9	91.1	87.9	85.9	75.4	84.0
DIAL-it1	52.8	53.8	**5.5**	70.3	69.3	40.6	79.5	76.0	58.3	87.4	**77.4**	68.7	76.5	71.4	51.5	72.0
DIAL-it2	53.8	**51.3**	8.2	67.3	65.3	34.7	76.2	71.6	51.3	90.1	86.5	79.2	74.0	69.1	47.2	70.0
DIAL-it3	**52.5**	60.6	**5.5**	**64.0**	**64.4**	**28.0**	**68.1**	**59.5**	**33.9**	**84.5**	78.9	**67.3**	**69.5**	**63.9**	**37.6**	**60.0**

Table 3: Correlation with chatbot issue rates and KL divergence of issue category distributions, computed across 14 diverse chatbot configurations. Aggregate combines UserSim-160K-SFT, DIAL-it1, it2, and it3. Best values are in **bold**, second best are underlined.

Simulator	r	p	KL Div.
Zero-shot			
GPT-4o	.484	.079	1.566
Kimi-K2	.564	.036	.453
Llama-3.3-70B	.629	.016	.542
UserSim			
160K-SFT	.528	.052	.671
160K-DIAL-it1	.685	.007	.630
160K-DIAL-it2	**.818**	**.0003**	.318
160K-DIAL-it3	.781	.001	**.148**
Aggregate	.801	.0006	.152

Table 4: MultiWOZ 2.1 user simulator evaluation. Diversity metrics computed on user utterances against the human test set. Success rates are reported using the ConvLab-3 evaluator. Best fine-tuned values are in **bold**.

Method	TTR	D-2	D-3	MAUVE	Success (%)
Human	.037	.229	.470	–	–
RulePolicy	.004	.015	.026	.039	50.2
TUS	.007	.028	.065	.158	48.8
GenTUS	.010	.039	.086	.175	59.1
Zero-shot	.028	.150	.316	.071	99.1*
SFT	.022	.106	.228	.137	**98.3***
DIAL-it1	.022	.101	.212	.219	95.9*
DIAL-it2	.025	.119	.249	.251	94.7*
DIAL-it3	**.031**	**.211**	**.458**	**.390**	93.0*

*ConvLab-3 does not verify correctness for LLM agents.

biases. This confirms that domain-specific fine-tuning is a necessary foundation for realistic simulation, consistent with the broader literature on domain adaptation. However, while SFT achieves strong behavioral alignment, it suppresses lexical diversity through mode collapse. DIAL recovers this diversity while maintaining behavioral fidelity, producing simulators that can deceive increasingly sophisticated discriminators, demonstrating that realism requires both distributional accuracy and surface-level variation. Discriminator accuracy and human judgment directly quantify distributional realism by testing whether conversations are distinguishable from real interactions.

7 Conclusion

This work presents DIAL (Direct Iterative Adversarial Learning), a framework for creating realistic user simulators that can reliably expose failure modes in multi-turn dialogue systems. Applied to mental health support, DIAL shows that domain-specific fine-tuning combined with iterative adversarial refinement produces simulators that closely replicate real user behavior across linguistic, semantic, and behavioral dimensions.

The key contributions of our work are: (1) the DIAL methodology, which uses discriminator-derived preference pairs with DPO instead of policy gradient methods, avoiding the training instability of prior adversarial dialogue approaches while recovering lexical diversity suppressed by supervised learning and improving distributional alignment; (2) demonstration that DIAL enables accurate identification of diverse failure modes in the mental health support domain; and (3) comprehensive validation showing that DIAL-trained simulators reliably predict real model performance.

These results show that DIAL is a promising methodology for creating realistic user simulators in multi-turn dialogue systems. By enabling rapid iteration and systematic exploration of system failure modes before deployment, such simulators can reduce development time, protect users from sub-optimal experiences, and offer safe environments for RL-based optimization of conversation quality.

Limitations

While our approach demonstrates significant improvements in simulator realism, several limitations remain that warrant future investigation:

Mode Collapse: Although DIAL substantially improves lexical diversity, fine-tuned simulators still exhibit lower phrase-level variety than real users. Techniques such as controlled generation or mixture-of-experts architectures may further enhance diversity preservation.

Domain Transfer: While we provide initial validation on MultiWOZ 2.1 (Section 5.3), the primary evaluation focuses on mental health support. Extending DIAL training to transactional TOD benchmarks, as well as other complex domains such as tutoring and technical support, remains an area for future investigation.

End-to-End TOD Training and Evaluation: Although the current simulators enable valid offline evaluation of chatbot failure modes, their application for training and evaluating end-to-end dialogue policies remains an area for future investigation. It is hypothesized that increasing simulator realism may not necessarily improve raw task success rates when evaluated by the simulator itself. Instead, success rates are expected to become more calibrated with real-user outcomes, as less realistic simulators often exhibit excessive compliance, leading to inflated success metrics. For instance, Cheng et al. (2022) trains a user simulator with RL using task success as the reward, which encourages the simulator to become more compliant, contrary to the goal of surfacing system failures. Rigorous validation of this hypothesis requires access to real users for ground-truth comparison, a resource that is challenging to obtain at scale for most TOD benchmarks.

Ethical Considerations

All data used in this research were collected from real user conversations with the mental health chatbot, with explicit user consent. Personally identifiable information was removed through automated script-based detection and removal of names, phone numbers, email addresses, and other identifiers in accordance with data protection policies and ethical guidelines for human subjects research. However, due to the sensitive nature of mental health data, the datasets used in this research cannot be shared, despite the intention to promote reproducibility and open science.

Realistic user simulators could potentially be misused to train systems optimized for engagement rather than user well-being, or to manipulate vulnerable populations. The simulators developed in this work are intended for system evaluation and development, enabling rapid iteration without exposing users to suboptimal experiences. It should be noted that simulators trained on real conversations may perpetuate biases present in the original dataset, potentially disadvantaging certain demographic groups. Simulators should therefore be deployed with appropriate oversight and ethical review. Additionally, while simulators can reduce the need for human testing in many scenarios, they should complement rather than entirely replace human feedback for validating novel interventions.

Study procedures were approved as exempt by an Institutional Review Board.

References

Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, and 149 others. 2025. Kimi k2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ – a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Qinyuan Cheng, Linyang Li, Guofeng Quan, Feng Gao, Xiaofeng Mou, and Xipeng Qiu. 2022. Is MultiWOZ a solved task? an interactive TOD evaluation framework with user simulator. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1248–1259, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sam Davidson, Vidhi Raina, Houda Bouamor, and Bill Byrne. 2024. Prompting large language models for user simulation in task-oriented dialogue systems. *Computer Speech and Language*, 89:101697.

Wenjie Dong, Sirong Chen, and Yan Yang. 2025. ProTOD: Proactive task-oriented dialogue system based on large language model. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING 2025)*, pages 9147–9164, Abu Dhabi, UAE.

Layla El Asri, Jing He, and Kaheer Suleman. 2016. A sequence-to-sequence model for user simulation in spoken dialogue systems. In *Interspeech 2016*, pages 1151–1155.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron

Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Joey Hong, Sergey Levine, and Anca Dragan. 2023. Zero-shot goal-directed dialogue via RL on imagined conversations. *Preprint*, arXiv:2311.05584.

Joey Hong, Jessica Lin, Anca Dragan, and Sergey Levine. 2024. Interactive dialogue agents via reinforcement learning on hindsight regenerations. *Preprint*, arXiv:2411.05194.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, and 399 others. 2024. GPT-4o system card. *Preprint*, arXiv:2410.21276.

Simon Keizer, Caroline Dockes, Norbert Braunschweiger, Svetlana Stoyanchev, and Rama Doddipatla. 2023. Adversarial learning of neural user simulators for dialogue policy optimisation. *Preprint*, arXiv:2306.00858.

Florian Kreyssig, Iñigo Casanueva, Paweł Budzianowski, and Milica Gašić. 2018. Neural user simulation for corpus-based policy optimisation of spoken dialogue systems. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 60–69, Melbourne, Australia. Association for Computational Linguistics.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2157–2169, Copenhagen, Denmark. Association for Computational Linguistics.

Zekun Li, Wenhu Chen, Shiyang Li, Hong Wang, Jing Qian, and Xifeng Yan. 2022. Controllable dialogue simulation with in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4330–4347, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hsien-chin Lin, Christian Geishauser, Shutong Feng, Nurul Lubis, Carel van Niekerk, Michael Heck, and Milica Gasic. 2021. Domain-independent user simulation with transformers for task-oriented dialogue systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 445–456, Singapore and Online. Association for Computational Linguistics.

Hsien-chin Lin, Christian Geishauser, Shutong Feng, Nurul Lubis, Carel van Niekerk, Michael Heck, and Milica Gasic. 2022. GenTUS: Simulating user behaviour and language in task-oriented dialogues with generative transformers. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 270–282, Edinburgh, UK. Association for Computational Linguistics.

Bing Liu and Ian R. Lane. 2017. Iterative policy learning in end-to-end trainable task-oriented neural dialog models. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 482–489.

Xiang Luo, Zhiwen Tang, Jin Wang, and Xuejie Zhang. 2024. DuetSim: Building user simulator with dual large language models for task-oriented dialogues. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5414–5424, Torino, Italia. ELRA and ICCL.

Hyunji Nam, Omer Gottesman, Amy Zhang, Dean Foster, Emma Brunskill, and Lyle Ungar. 2025. Efficient RL for optimizing conversation level outcomes with an LLM-based tutor. *Preprint*, arXiv:2507.16252.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744.

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

José Pombal, Maya D'Eon, Nuno M. Guerreiro, Pedro Henrique Martins, António Farinhas, and Ricardo Rei. 2025. MindEval: Benchmarking language models on multi-turn mental health support. *Preprint*, arXiv:2511.18491.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve J. Young. 2009. The hidden agenda user simulation model. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4):733–747.

Jost Schatzmann, Karl Weilhammer, Matthew N. Stuttle, and Steve J. Young. 2006. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The Knowledge Engineering Review*, 21(2):97–126.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Ivan Sekulić, Silvia Terragni, Victor Guimarães, Nghia Khau, Bruna Guedes, Modestas Filipavicius, André Ferreira Manso, and Roland Mathis. 2024. 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)*, pages 19–35, St. Julians, Malta. Association for Computational Linguistics.

Lior Shani, Aviv Rosenberg, Asaf Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Rémi Munos. 2024. Multi-turn reinforcement learning from preference human feedback. *Preprint*, arXiv:2405.14655.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 464 others. 2025. Openai GPT-5 system card. *Preprint*, arXiv:2601.03267.

Charlie Snell, Ilya Kostrikov, Yi Su, Mengjiao Yang, and Sergey Levine. 2023. Offline RL for natural language generation with implicit language Q learning. In *International Conference on Learning Representations*.

Yuda Song, Gokul Swamy, Aarti Singh, J. Andrew Bagnell, and Wen Sun. 2024. The importance of online data: Understanding preference fine-tuning via coverage. In *Advances in Neural Information Processing Systems*, volume 37.

Fahim Tajwar, Anikait Singh, Archit Sharma, Rafael Rafailov, Jeff Schneider, Tengyang Xie, Stefano Ermon, Chelsea Finn, and Aviral Kumar. 2024. Preference fine-tuning of LLMs should leverage suboptimal, on-policy data. In *Forty-first International Conference on Machine Learning*.

Ryuichi Takanobu, Runze Liang, and Minlie Huang. 2020. Multi-agent task-oriented dialog policy learning with role-aware reward decomposition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 625–638, Online. Association for Computational Linguistics.

Siddharth Verma, Justin Fu, Sherry Yang, and Sergey Levine. 2022. CHAI: A CHatbot AI for task-oriented dialogue with offline reinforcement learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4471–4491, Seattle, United States. Association for Computational Linguistics.

Heng-Da Xu, Xian-Ling Mao, Puhai Yang, Fanshu Sun, and Heyan Huang. 2024. Rethinking task-oriented dialogue systems: From complex modularity to zero-shot autonomous agent. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2024. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *Preprint*, arXiv:2406.12045.

Xiao Yu, Maximillian Chen, and Zhou Yu. 2023. Prompt-based Monte-Carlo tree search for goal-oriented dialogue policy planning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7101–7125, Singapore. Association for Computational Linguistics.

Xiangyu Zhao, Long Xia, Lixin Zou, Hui Liu, Dawei Yin, and Jiliang Tang. 2021. UserSim: User simulation via supervised generative adversarial network. In *Proceedings of the Web Conference 2021*, pages 3582–3589.

Zihui Zhao and Zechang Li. 2025. Reflective preference optimization (RPO): Enhancing on-policy alignment via hint-guided reflection. *Preprint*, arXiv:2512.13240.

Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. 2024. ArCHer: Training language model agents via hierarchical multi-turn RL. In *Proceedings of the 41st International Conference*

on Machine Learning, volume 235 of *Proceedings of Machine Learning Research*, pages 62178–62209. PMLR.

Qi Zhu, Christian Geishauser, Hsien-chin Lin, Carel van Niekerk, Renato Barahona, Shutong Feng, Michael Heck, Xiaochen Feng, Yue Su, and Milica Gasic. 2023. ConvLab-3: A flexible dialogue system toolkit based on a unified data format. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 91–102, Singapore. Association for Computational Linguistics.

Ziyi Zhu, Olivier Tieleman, Alexey Bukhtiyarov, and Jinghong Chen. 2026. CyclicJudge: Mitigating judge bias efficiently in LLM-based evaluation. *Preprint*, arXiv:2603.01865.

A Context Construction for Mental Health Support Chatbot

This appendix describes the specific context structure used for our mental health support chatbot user simulator. While the general principles (role definition, user background, conversational history, session grounding) apply broadly to TOD systems, the specific implementation reflects the requirements of mental health support.

Our context construction follows a hierarchical structure with the following components:

System Instruction: Explicit instructions defining the simulator's role as a user in mental health support conversations, including guidelines for consistency and realistic behavior.

User Profile: Demographics and onboarding responses establishing presenting concerns, goals, and background information. This captures the user's initial state and motivations for seeking support.

Long-term Context: Aggregated information about the user's mental health support journey, including recurring themes, progress over time, and persistent concerns that span multiple sessions. This enables the simulator to maintain consistency across extended timeframes.

Previous Session Summaries: Summaries of prior sessions capturing key topics discussed, insights gained, and mental health support interventions attempted. This provides cross-session continuity without requiring the full message history.

Current Session Summary: A summary of the current session being simulated, extracted from the original conversation. This grounds the simulator in the specific topics and emotional tone of the target session.

Raw Message History: Message-by-message conversation history from recent sessions, providing detailed behavioral context and immediate conversational flow.

The complete context is truncated to 15,000 tokens, prioritizing recent session content and current session summary over distant history. All context components are automatically generated using production usage, with approximately 100,000 user profiles available for training and evaluation. This enables our approach to scale naturally with system usage without requiring manual annotation or synthetic data generation.

B Detailed Analysis

This appendix provides detailed visualizations and analysis supporting the main results, including embedding visualizations, correlation analysis, and action distribution comparisons.

B.1 Embedding Visualizations

We embed all sessions using OpenAI's text-embedding-3-large, including up to 50 messages per session within the model's token limit. We compute t-SNE projections using `scikit-learn`[1] with perplexity 30 on all simulated and real sessions, then visualize 50 randomly sampled sessions for clarity.

Figure 3 shows the t-SNE visualization of session embeddings. Each base model (GPT-4o, Kimi-K2-Instruct, Llama-3.3-70B-Instruct) forms a distinct cluster (circles) separate from real sessions (stars), indicating systematic behavioral differences. In contrast, fine-tuned simulators (squares) cluster tightly around their corresponding real sessions, demonstrating that rich context grounding enables simulators to maintain topical coherence and behavioral fidelity.

B.2 MAUVE Configuration

MAUVE scores are computed using `mauve-text`[2]. Utterances are embedded with OpenAI's text-embedding-3-small model, quantized into $k = 250$ clusters via k-means, and the divergence frontier area is computed following Pillutla et al. (2021). For the mental health domain, all user utterances from a simulator's sessions form distribution P, while all real user utterances form distribution Q.

[1] https://scikit-learn.org/
[2] https://github.com/krishnap25/mauve

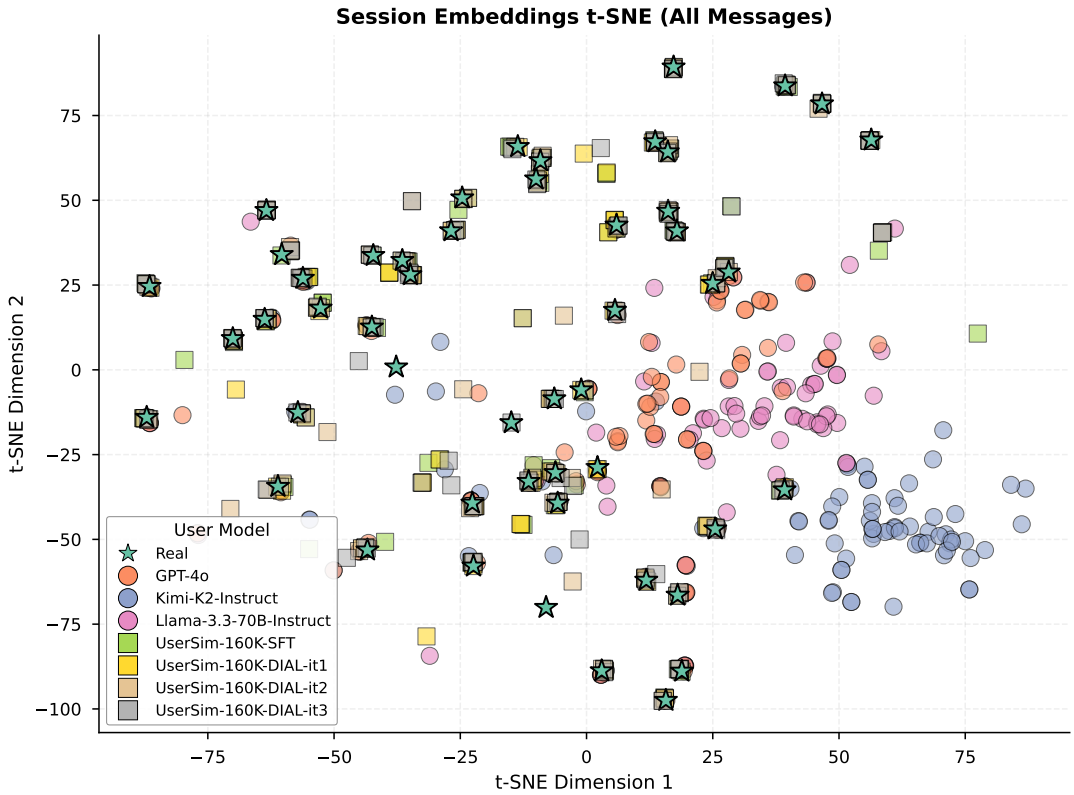

Figure 3: t-SNE visualization of session embeddings. Production sessions are marked with stars.

The same procedure is applied to MultiWOZ experiments using user utterances from the test set as Q.

B.3 Issue Category Distribution

Figure 4 shows the discrepancy between simulated and real session issue category distributions. The simulator over-represents certain categories, including "Chatbot contradicts itself", "Chatbot forgets in context", and "Unclear or confusing wording"—likely because simulated users engage more persistently with problematic chatbot behaviors, triggering repeated issues. Conversely, the simulator under-represents categories like "Duplicate chatbot messages", "Chatbot hallucinates", and "Premature wrap-up", which require specific conversational histories that may be less common in simulation. The relatively small magnitude of most discrepancies (within ± 2.5 percentage points) indicates reasonable distributional alignment overall.

B.4 User Action Distributions

Figure 5 compares user action distributions across simulators. For negative indicators (left), zero-shot models dramatically underestimate rates of challenging behaviors: GPT-4o shows near-zero "Aggressive to Chatbot", and all base models under-represent "Asks for Clarity" and "Disagrees with Chatbot." This confirms that zero-shot models produce unrealistically compliant users.

For positive indicators (right), zero-shot models exhibit the opposite bias, substantially overrepresenting "Change Talk" and "Agrees with Chatbot." This creates an artificially positive mental health support environment that fails to challenge the chatbot or surface issues. Fine-tuned simulators achieve close alignment with real sessions across both indicator types, with DIAL iterations showing incremental improvements in matching real distributions.

We use Binary Cross-Entropy (BCE) to measure distributional alignment:

$$\text{BCE}(p, q) = -\sum_i [p_i \log(q_i) + (1-p_i) \log(1-q_i)] \tag{6}$$

where p is the real distribution and q is the simulated distribution. BCE is appropriate for such multi-label distributions where action probabilities do not sum to 1, while KL divergence requires proper probability distributions.

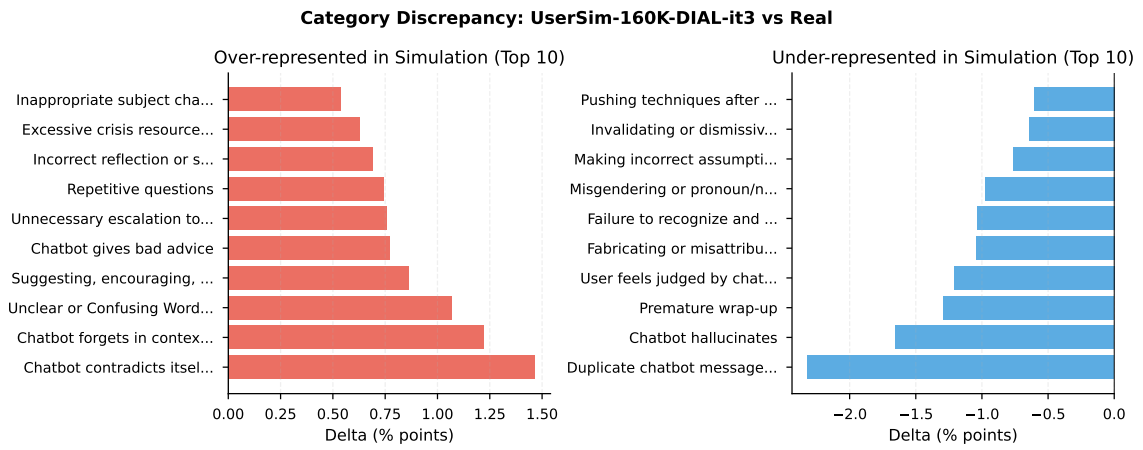

Figure 4: Issue category discrepancy between UserSim-160K-DIAL-it3 and real conversations. Deltas shown as percentage point differences.

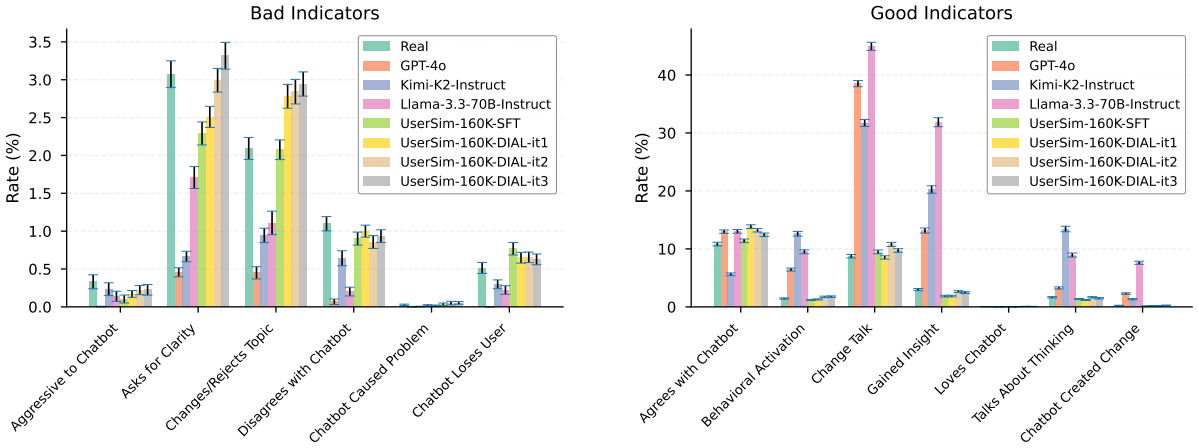

Figure 5: User action distributions comparing simulators to real data. Error bars represent standard error.

B.5 Correlation with Production

Figure 6 shows the correlation between simulated and real issue rates across 14 diverse chatbot configurations. The Pearson correlation coefficient $r = 0.818$ indicates a strong linear relationship, with the simulator correctly ranking model quality. The p-value represents the probability of observing this correlation by chance under the null hypothesis of no correlation. It is calculated using a t-test: $t = r\sqrt{(n-2)/(1-r^2)}$ with $n - 2$ degrees of freedom.

DIAL substantially improves correlation with real issue rates, as shown in Table 3. However, we note that several models achieve strong correlations even without DIAL. This suggests that for rolled-out evaluation or RL training applications, ensembling multiple user simulator models could reduce noise and improve stability. The plot shown here highlights the best-performing model (UserSim-160K-DIAL-it2) to demonstrate the upper bound of achievable predictive validity.

B.6 A/B Test Replication

To validate predictive utility beyond aggregate correlation, we replicate a recent A/B test comparing control chatbot models against treatment versions. The treatment models were fine-tuned using DPO on synthetic issue data generated by our simulator, with the goal of reducing issue rates. We selected this experiment because it directly targets issue rates and produced statistically significant results that provide a clear benchmark for replication.

The intervention ultimately failed to achieve its intended goal, with the fine-tuned models actually increasing issue rates rather than reducing them. This negative result makes it an ideal test case: a valid simulator should correctly identify that the fine-tuning applied here is ineffective. Table 6 shows that all three models exhibited consistent

Table 5: BCE loss from real user action distributions by category. Best values are in **bold**, second best are underlined.

Simulator	**All Actions**	Bad Indicators	Good Indicators	Moderators	User Features
Zero-shot					
GPT-4o	.1866	.0877	.1857	.1064	.2952
GPT-5.2	.2547	.0701	.3116	.1046	.3989
Kimi-K2	.1912	.0645	.2063	.1041	.3095
Qwen-2.5-7B	.1735	.0692	.1686	.1078	.2867
Llama-3.3-70B	.2072	.0627	.2455	.1047	.3221
UserSim					
160K-SFT	.1587	.0599	.1368	.1047	**.2806**
160K-DIAL-it1	.1596	.0598	.1373	.1043	.2833
160K-DIAL-it2	**.1586**	**.0597**	.1370	**.1040**	.2807
160K-DIAL-it3	.1587	**.0597**	**.1366**	.1041	.2811

directional trends between simulation and real data.

C Model and Training Details

C.1 Model Licenses and Terms of Use

All models used in this work are distributed under licenses permitting research use. Llama-3.3-70B-Instruct and Llama-3.1-8B-Instruct are released under the Llama 3.3 and Llama 3.1 Community License Agreements respectively, which permit commercial and research use with certain restrictions.

C.2 Training Hyperparameters

Table 7 shows the hyperparameters used for SFT of the base user simulator and DPO during DIAL iterations.

C.3 Computational Budget and Infrastructure

All experiments were conducted on a computing infrastructure consisting of 8×NVIDIA H100 GPUs. The base user simulator (SFT) required approximately 55 GPU-hours of training time. Each DIAL iteration involved discriminator training and DPO refinement, requiring approximately 2.5 GPU-hours per iteration. With 10 DIAL rounds across all experiments, the total computational budget was approximately 80 GPU-hours (55 hours for SFT + 25 hours for DIAL iterations).

D Issue Analyzer Validation

To assess the reliability of our LLM-based issue analyzer, we conducted a human evaluation study on 100 randomly sampled real-user sessions. For each session, a human annotator reviewed all issues flagged by the model and indicated whether they agreed with each detected issue. The annotator also independently reviewed the full session to identify any issues the model missed. We evaluated three frontier LLMs: GPT-5, Claude-Sonnet-4-5, and Gemini-3-Pro.

Across the 100 sessions, the human annotator identified a total of 49 ground-truth issues. We report two metrics: **Agreement Rate** (precision), the percentage of model-detected issues with which the human annotator agreed, and **Omission Rate** ($1 - \text{recall}$), the percentage of human-identified issues that the model failed to detect.

As shown in Table 8, Claude-Sonnet-4-5 achieves the highest agreement rate, while GPT-5 achieves the lowest omission rate with the second-highest agreement, making it a suitable choice for comprehensive issue detection where recall is prioritized. False positives were concentrated in subjective categories such as "Unempathetic responses" and "Chatbot is sycophantic," where annotator and model thresholds can reasonably differ. Omitted issues tended to involve subtle conversational dynamics (e.g., gradually circular conversations) that require tracking patterns across many turns. These results support the use of GPT-5 as the primary issue annotation model given its low omission rate, though they also highlight that LLM-based issue detection remains an imperfect proxy for human judgment, particularly for nuanced or ambiguous issue categories.

E User Action Categories

This appendix provides example user action categories from our taxonomy used to annotate and analyze user behavior in mental health support conversations. Actions are classified as positive mental health indicators (Good), negative indicators (Bad), or user features that should change over time. The full taxonomy contains 20+ categories; we highlight key examples below.

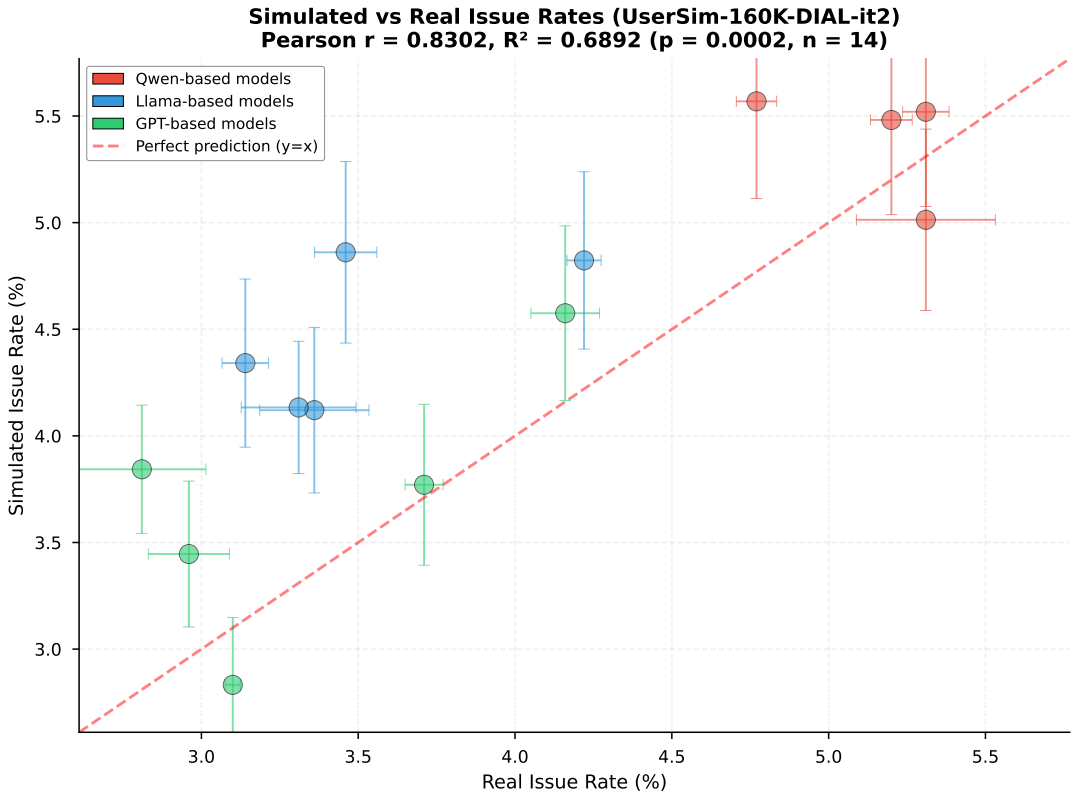

Figure 6: Correlation between simulated and real issue rates for UserSim-160K-DIAL-it2 across 14 chatbot model configurations spanning multiple model families (Qwen-based, Llama-based, GPT-based). Error bars represent standard error.

E.1 Example Positive Indicators

- **Gained an insight:** User expresses that they learned something new about themselves or about their situation.
- **Change talk:** User uses change talk to set a task goal or commit to a healthy action—something specific.
- **Agrees with chatbot:** User agrees with an observation, suggestion, or advice that the chatbot gives.

E.2 Example Negative Indicators

- **Asks for clarity:** User asks for the chatbot to clarify what it's doing or saying.
- **Aggressive to chatbot:** User gets angry with the chatbot or is combative or abusive in some way.
- **Chatbot loses user:** User is not sure how to respond to the chatbot and responds in a way that suggests they're disengaging.

E.3 Example User Features

- **General distress:** User reports discomfort or suffering that is vague and lacks a specific object or expresses not knowing who they are or what life means to them.
- **Negative affect:** User is able to acknowledge a negative emotion and make it explicit—a specific emotion must be identified; general expressions of negative affect fall under general distress.
- **Positive affect:** User expresses a positive emotion and makes it explicit—a specific emotion must be identified.

F Issue Categories

This appendix provides example issue categories from our taxonomy used to evaluate chatbot quality. The full taxonomy contains 35+ categories organized into six main categories (User Distress, User Confusion, User Frustration, User Losing Trust, Chatbot Violates Policies, App Bugs); we highlight key examples below.

Table 6: Issue rate comparison between simulated and real evaluations with two-tailed significance tests ($\alpha = 0.05$). The treatment (fine-tuned with DPO on synthetic issue data to reduce issues) failed and actually increased issue rates. Both simulation and real data show consistent directional trends, validating the simulator's predictive utility.

	Simulation (UserSim-160K-SFT)				Production			
Model	Control	Treatment	z	p	Control	Treatment	z	p
Model A (Qwen-based)	2.65%	3.25%	2.83	0.005*	2.71%	3.56%	3.27	0.001*
Model B (Llama-based)	2.42%	2.87%	2.23	0.026*	2.32%	2.64%	1.35	0.178
Model C (GPT-based)	1.78%	2.13%	2.02	0.044*	1.54%	1.42%	-0.66	0.511

*Statistically significant at $\alpha = 0.05$

Table 7: Hyperparameters for supervised fine-tuning (SFT) and Direct Preference Optimization (DPO).

Hyperparameter	**SFT**	**DPO**
Epochs	1	1
Batch size	1	1
DPO beta (β)	–	1
LoRA rank	8	8
LoRA alpha	16	16
LoRA trainable modules	all-linear	all-linear
Learning rate	5×10^{-5}	1×10^{-5}
Learning rate scheduler	constant	constant
Warmup ratio	0	0.1
Max gradient norm	1	1
Weight decay	0	0

Table 8: Human validation of LLM-based issue analyzers. Detected denotes the number of issues flagged by each model. Best values are in **bold**, second best are underlined.

Model	**Agreement (%)**	**Omission (%)**	**Detected**
GPT-5	45.2	**14.3**	93
Claude-Sonnet-4-5	**63.6**	42.9	44
Gemini-3-Pro	38.0	22.4	100

F.1 User Distress Examples

- **Invalidating or dismissive framing:** The chatbot uses language that minimizes the user's struggle or suggests they could simply choose to feel better (e.g., framing complex emotional states as mere choices).

- **Unempathetic responses:** The chatbot responds in ways that lack empathy in distressing or painful situations, including being overly brief, blunt, cold, detached, or otherwise failing to appropriately acknowledge or validate the user's emotional state.

- **Premature wrap-up:** The chatbot initiates ending the session or wraps up the conversation when the user has explicitly expressed wanting to continue; it's clear the user doesn't want to end; the user is mid-sentence; or at otherwise inappropriate moments, such as immediately after the user has shared something sensitive or vulnerable.

F.2 User Confusion Examples

- **Chatbot hallucinates:** The chatbot invents or misremembers information about the user—such as their experiences, feelings, or history—and the user explicitly corrects or objects to the inaccuracy.

- **Unclear or confusing wording:** The chatbot says something that is unusually complex, convoluted, or strange, making it difficult for the user to understand the intended meaning or follow the flow of conversation.

- **Chatbot contradicts itself:** The chatbot makes statements or suggestions that directly contradict each other in meaning, tone, or advice, creating clear confusion or inconsistency that cannot reasonably be explained as exploring multiple perspectives.

F.3 User Frustration Examples

- **Circular conversations:** The chatbot leads the conversation in circles, revisiting the same topics or lines of discussion that have already been addressed, without adding new insight or progress. The issue applies only when the user shows clear signs of frustration or impatience with this repetition.

- **Ignoring user's request to end session:** When the user clearly indicates they need to leave or pause or end the conversation (explicitly or implicitly), the chatbot ignores this and continues asking questions or advancing the discussion instead of acknowledging and closing promptly.

- **Repetitive questions:** The chatbot repeats a question or statement that is nearly identical

to something it already said earlier in the same session, without any clear reason or therapeutic justification for the repetition.

F.4 User Losing Trust Examples

- **Chatbot gives bad advice:** The chatbot gives advice or suggestions that are objectively poor, unhelpful, or clearly irrelevant to the user's situation—not merely advice that the user disagrees with or finds unappealing.

- **Chatbot is sycophantic:** The chatbot responds with excessive, exaggerated, or insincere praise or agreement that goes beyond normal empathy, validation, or encouragement. The issue applies only when the flattery or agreement is so strong or uncritical that it feels unnatural, manipulative, or disingenuous.

F.5 Policy Violation Examples

- **Failure to follow crisis protocol:** The chatbot does not provide required crisis resources, encourage immediate human help, or include the mandated disclaimer about AI limitations when suicide or imminent harm is mentioned.

- **Providing medical diagnoses:** The chatbot gives specific medical or mental health diagnoses, which it is not qualified to provide as an AI system.

- **Inappropriate self-disclosure:** The chatbot makes personal or biographical claims that go beyond light, clearly artificial expressions of personality. The issue applies only when the chatbot refers to having real human experiences, relationships, or memories in a way that blurs its AI identity.

F.6 App Bug Examples

- **Incomplete message:** The chatbot's message is cut off or incomplete, typically due to streaming generation failure, leaving the response unfinished mid-sentence or mid-thought.

- **Transcription errors:** Speech-to-text conversion errors cause the chatbot to significantly misunderstand the user, leading to a noticeable impact on the conversation.