OmniSimulator: Aligning Small Language Models for Authentic Heterogeneous Behavior Modeling

Jiawei Chen[1,2], **Ruoxi Xu**[1,2], **Boxi Cao**[1], **Zhuoqun Li**[1,2], **Ruotong Pan**[3], **Yunfei Zhang**[3], **Zhirui Yang**[3], **Jingyan Chen**[3], **Tingting Gao**[3], **Han Li**[3], **Yaojie Lu**[1], **Xianpei Han**[1], **Le Sun**[1], **Xiangyu Wu**[3†], **Hongyu Lin**[1†]
[1] Chinese Information Processing Laboratory, Institute of Software, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences
[3] Kuaishou Technology
chenjiawei2024@iscas.ac.cn

Abstract

Large language models (LLMs) have shown great potential as general-purpose user simulators for interactive systems. Prior studies have demonstrated that training on high-quality data can substantially improve model reasoning and downstream task performance. However, the upper bound of user simulation is still constrained by the model's reasoning paradigm: native chain-of-thought often degenerates into shallow pattern matching and fails to adequately model the behavioral factors underlying real user decisions. In this paper, we propose ***OmniSimulator***, a novel training framework designed to improve the predictive ability of small LLMs for realistic user simulation on short-video platform environments. Specifically, OmniSimulator organizes behavior-oriented reasoning through a structured schema with five dimensions, carefully selects data from real user-behavior scenarios, and introduces a teacher model to generate high-quality reasoning traces under the proposed schema, helping small models learn behavior-relevant reasoning patterns rather than relying on superficial behavioral correlations. Based on this process, we construct 9,881 high-quality training instances from three months of historical interaction trajectories of 200 real users, averaging about 50 distilled examples per user. Experimental results show that OmniSimulator improves the average score of Qwen3-8B by 34.33%, achieves performance competitive with Claude-Sonnet-4.5, and supports smaller-scale, self-hosted deployment. These results suggest that learning structured behavior-oriented reasoning patterns is a promising direction towards scalable and high-fidelity LLM-based user simulation[1].

1 Introduction

Large language models (LLMs) have recently shown strong promise in a growing range of downstream applications [30, 13, 22, 12, 8], including recommender systems [5, 10, 7], advertising [20], customer service [27, 14], and interactive agents [29, 26]. Beyond their generative ability, an increasingly appealing direction is to use LLMs as *general-purpose user simulators* [4, 19, 7], enabling offline evaluation, policy testing, product optimization, and behavior modeling at a scale that would otherwise require expensive online experiments [18, 19]. This trend is especially important for short-video platforms, where user behavior is inherently long-horizon, cross-scenario, and heterogeneous [9]: a user may browse videos, enter live rooms, click ads, shop in e-commerce, search for information, and interact with customer service within one unified behavioral trajectory. Recent studies have

†Corresponding authors.
[1]Our code is publicly available at https://github.com/icip-cas/OmniSimulator

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

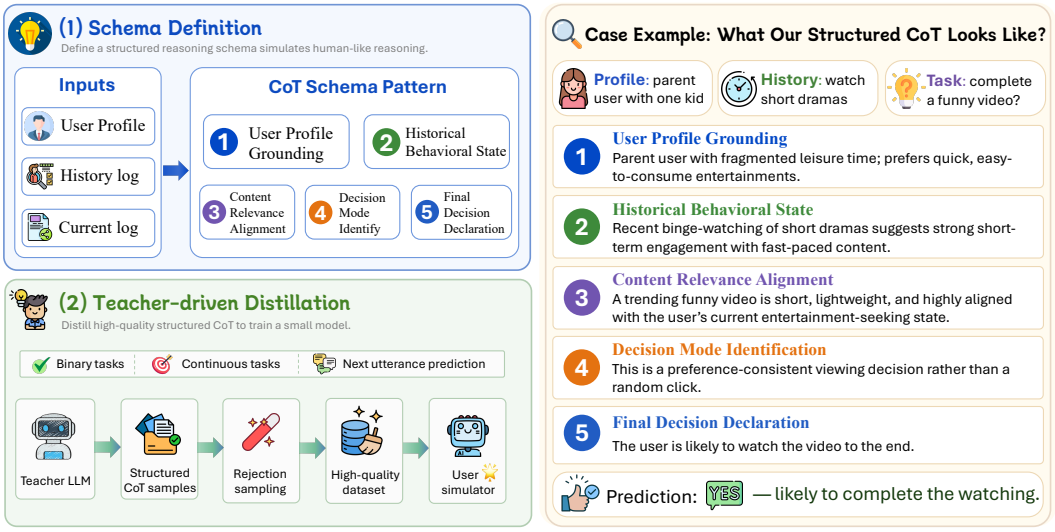


Figure 1: **Overview of OmniSimulator.** The framework consists of two stages. **(1) Schema Definition:** we define a structured chain-of-thought schema that guides the model to simulate human-like internal reasoning from three inputs: user profile, historical interaction logs, and the current scenario. The schema decomposes reasoning into five dimensions: *User Profile Grounding*, *Historical Behavioral State*, *Content Relevance Alignment*, *Decision Mode Identification*, and *Final Decision Declaration*. **(2) Teacher-driven Distillation:** a teacher LLM generates structured reasoning traces under the schema for different task types, and high-quality traces are selected through rejection sampling to construct the distilled training set for small user simulators. The right panel shows a case example of the structured user-state reasoning trace, where the model infers latent user state step by step and then predicts the final correct behavior.

shown that training on high-quality data can substantially improve reasoning ability and downstream task performance of LLMs, suggesting that post-training and supervision quality are critical for building stronger simulators. However, real user simulation remains far from solved. On challenging real-world benchmarks such as OmniBehavior [9], even frontier models achieve limited performance, while also exhibiting systematic biases such as overestimating positive actions, generating overly optimistic utterances, and collapsing different users into an average persona [6, 9]. Moreover, although closed-source models such as Claude-Opus-4.5 [2] provide stronger performance, their cost becomes prohibitive when conditioning on long interaction histories, making them difficult to deploy at industrial scale.

A difficulty of user simulation is that real behavior is not generated by direct surface matching between historical interactions and the current scenario. Instead, user actions are mediated by latent mental states, such as stable preference, recent interest drift, transient fatigue, emotional tendency, and momentary decision style. In other words, predicting whether a user will click, purchase, continue watching, or say a particular utterance requires the model to first recover *why this user is likely to act in that way at the current scenario*. This observation naturally suggests that reasoning capability counts. If the model can explicitly infer the hidden state underlying the next action, it should in principle make more faithful predictions than a model that only performs shallow behavioral imitation. Chain-of-thought (CoT) therefore appears to be a natural candidate for improving user simulation, since it provides an intermediate reasoning process between observed history and predicted behavior. However, our empirical analysis reveals that simply enabling native thinking mode brings almost no benefit for user simulation: across Qwen3 models, the performance of thinking and non-thinking settings is often nearly identical, and in some cases the thinking variant even performs worse. A closer inspection of native CoT outputs explains why: existing models rarely use reasoning to reconstruct the user's latent mental state; instead, they tend to produce generic justifications, shallow pattern matching, or direct guesses based on superficial correlations. This suggests that the bottleneck is not merely whether the model generates more reasoning tokens, but whether its reasoning follows a behavior-oriented structure.

Motivated by this, we propose **OmniSimulator**, a novel training framework designed to improve small LLMs for real user simulation on short-video platform environments. The key idea is to teach the model to follow behavior-oriented structured reasoning rather than free-form reasoning. Specifically, OmniSimulator introduces a structured CoT schema that forces the model to infer latent user states through five dimensions: *User Profile Grounding*, *Historical Behavioral State Inference*, *Content Relevance Alignment*, *Decision Mode Identification*, and *Final Decision Declaration*. On top of this schema, we build a task-aware causal distillation pipeline: a teacher model (GPT-OSS-120B [1]) generates multiple reasoning traces for each prediction point under task-specific prompting strategies, and only traces consistent with ground truth are retained through rejection sampling. Following this process, we construct 9,881 high-quality distilled training instances from three months of interaction histories of 200 real users. Experiments show that OmniSimulator substantially improves small models over zero-shot and free-CoT baselines, enables Qwen3-8B to achieve performance competitive with strong proprietary models such as Claude-Sonnet-4.5 [3], and alleviates overestimation bias while supporting smaller-scale deployment. These results suggest that the upper bound of LLM-based user simulation is constrained not only by model scale, but also by the reasoning supervision paradigm.

Our main contributions are summarized as follows:

- We identify a key weakness of existing LLM-based user simulators: native CoT struggles to predict realistic user behavior, yielding only marginal improvements in behavioral prediction over non-thinking settings.
- We propose OmniSimulator, a novel training framework that combines a structured five-dimensional CoT schema with a task-aware causal distillation pipeline, enabling small models to learn behavior-oriented reasoning patterns for user simulation.
- We construct 9,881 high-quality distilled training instances from real-world long-horizon user trajectories, and OmniSimulator substantially improves small LLMs, achieves performance competitive with strong closed-source baselines and mitigates overestimation bias while supporting efficient small-model deployment.

2 Methodology

In this section, we first formulate user behavior simulation on short-video platforms, and then present **OmniSimulator**, our training framework for improving small LLMs on realistic user simulation. The central idea is that real user behavior is not generated by direct surface matching between historical interactions and the current scenario. Instead, it is mediated by a latent decision process shaped by stable preference, recent interest drift, transient affect, and the decision regime at the current moment. Existing LLMs, even with native thinking mode, often fail to reconstruct this hidden process and instead rely on shallow behavioral priors. To address this issue, OmniSimulator explicitly teaches the model to infer latent user states before predicting behavior. The framework consists of two tightly coupled components: (1) a *structured chain-of-thought schema* that organizes reasoning into a fixed sequence of human-like internal deliberation steps, and (2) a *task-aware rationale distillation pipeline* that converts teacher-generated reasoning traces into high-quality supervision for student models.

2.1 Problem Formulation

We formulate user simulation as a user-conditioned prediction problem over long-horizon, cross-scenario interaction trajectories. For a user u, let

$$H_t^u = \{e_1^u, e_2^u, \ldots, e_t^u\} \tag{1}$$

denote the historical interaction sequence before time step $t+1$, where each event e_i^u may come from different scenarios such as video browsing, live streaming, e-commerce shopping, customer-service dialogue, advertisement recommendation, and search. Let x_{t+1}^u denote the current scenario context. The goal is to predict the corresponding user response

$$y_{t+1}^u = f_\theta(H_t^u, x_{t+1}^u), \tag{2}$$

where y_{t+1}^u may be a binary decision, a continuous value, or a natural-language utterance.

A direct mapping from (H_t^u, x_{t+1}^u) to y_{t+1}^u is often insufficient. In realistic platforms, a user's next behavior is mediated by hidden mental states that are not explicitly recorded in logs, such as stable

preference, short-term tendency, fatigue, emotion, and momentary decision style. We therefore introduce an implicit latent state z_t^u and decompose user simulation as

$$p(y_{t+1}^u \mid H_t^u, x_{t+1}^u) = \sum_{z_t^u} p(y_{t+1}^u \mid z_t^u, x_{t+1}^u)\, p(z_t^u \mid H_t^u). \tag{3}$$

Since z_t^u is unobserved, OmniSimulator does not supervise it as an explicit label. Instead, we represent it as a structured reasoning trace in natural language, and train the model to infer this trace before answering the final behavior prediction.

2.2 Preliminary Analysis: Revealing Native Thinking Effects

We first investigate whether native thinking mode alone is sufficient for realistic user simulation. Intuitively, this task appears to benefit from reasoning, since the model must recover the hidden factors that connect long interaction histories to current decisions. However, our preliminary analysis reveals that this intuition does not hold under standard prompting.

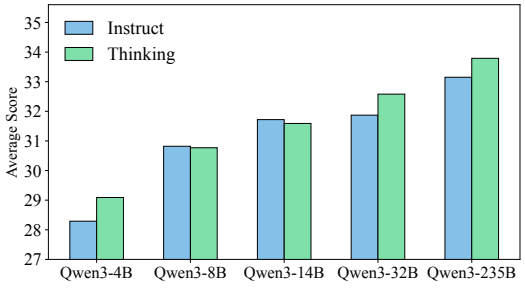


Figure 2: Preliminary analysis on native thinking mode. Across Qwen backbones, enabling native thinking yields only marginal or inconsistent gains over standard instruction-following inference.

Figure 2 compares the average OmniBehavior scores of multiple Qwen3-series [28] backbones under two settings: INSTRUCT and THINKING. We observe that enabling native thinking yields little or no consistent gain over standard instruction-following inference. Across model scales, the two settings are often nearly identical, and in some cases the thinking variant even performs worse. This result suggests that the bottleneck is not the absence of additional reasoning tokens, but the absence of the *right reasoning structure*. By inspecting native CoT outputs, we find that models often produce generic justifications or shallow guesses, failing to explicitly reconstruct the user's latent psychological state. This observation motivates our approach: rather than simply allowing the model to think more, we guide it to follow a structured, behavior-oriented reasoning process.

2.3 Overview of OmniSimulator

Figure 1 illustrates the overall framework. For each prediction point, we first provide the teacher model with the user's historical interaction trajectory and the current scenario context. The teacher is required to reason under a structured schema that explicitly reconstructs the user's latent mental state. To increase diversity, we sample multiple reasoning traces for each example with stochastic decoding. Depending on the task type, we apply different prompting strategies to control the degree of access to ground truth. The resulting traces are filtered by rejection sampling, and only those whose final answers are consistent with the observed outcome are retained as supervised fine-tuning data. Finally, the student model is trained to generate both the structured reasoning trace and the final prediction.

This design follows a simple principle: rather than only teaching the model what to predict, we also provide structured rationale supervision for how the prediction may be derived from behavioral evidence. In this way, latent-state reconstruction becomes explicit, learnable, and distillable.

Structured Chain-of-Thought Schema. The first stage of OmniSimulator is a structured CoT schema tailored for user simulation. Our design is motivated not only by the empirical analysis above, but also by cognitive theories of human behavior. Prior work suggests that behavior is mediated by stable person-level characteristics, dynamically activated cognitive-affective states, and their interaction with situational cues [17]. In parallel, dual-process accounts further distinguish between reflective, deliberative decision making and more impulsive behavior pathways [21]. Native thinking mode alone is therefore insufficient, because it leaves this behavioral reasoning path under-specified for human behavior simulation tasks. To address this issue, we require the model to generate reasoning within a `<think>` block following five fixed dimensions:

1. **User Profile Grounding:** The model first interprets stable user characteristics reflected in long-term history, such as age, gender, educational background. This step provides the global behavioral background of the user.
2. **Historical Behavioral State:** The model then infers the user's dynamic internal state from recent and long-horizon interactions. This includes long-term interest continuation, short-term preference drift, current fatigue level, and emotional tendency. Compared with the previous step, the focus here is not the user's general profile, but the user's state from the history sequences at the current moment.
3. **Content Relevance Alignment:** Next, the model matches the inferred user state with the current scenario and content. This step examines whether the current item is aligned with the user's active interests, intent, mood, or aversion, and serves as the bridge between historical state and prediction item.
4. **Decision Mode Identification:** The model further determines whether the current behavior is primarily driven by *rational decision* or *random decision*. Rational decisions are mainly explained by interpretable factors such as fatigue levels in this session, while random decisions are more influenced by curiosity or stochasticity. This distinction is essential for modeling negative outcomes and low-confidence actions.
5. **Final Decision Declaration:** Before closing the `</think>` block, the model should explicitly state its final decision, and this declaration is required to be fully consistent with the final output answer. This design reduces reasoning-answer mismatch and improves the controllability of the generated traces.

With this schema, OmniSimulator decomposes user simulation into the following process:

$$(H_t^u, x_{t+1}^u) \rightarrow r_t^u \rightarrow y_{t+1}^u, \tag{4}$$

where r_t^u denotes the structured reasoning trace. In practice, r_t^u serves as a textual proxy of the unobserved latent state z_t^u. Compared with unconstrained CoT, this schema forces the model to move from observable behaviors to hidden user states, and then from hidden user states to final actions.

Task-aware Rationale Distillation. Structured reasoning alone is not sufficient if teacher-generated traces are noisy. We therefore design a task-aware rationale distillation pipeline to obtain high-quality supervision for student training.

For each prediction item, we use a teacher model (GPT-OSS-120B [1] in our experiment) to rollout N candidate reasoning traces under the structured schema. Formally, given an example $(H_t^u, x_{t+1}^u, y_{t+1}^u)$, we sample

$$\mathcal{R}_t^u = \{r_{t,1}^u, r_{t,2}^u, \ldots, r_{t,N}^u\} \tag{5}$$

with relatively high temperature. This design allows the teacher to generate multiple candidate rationales for each prediction instance. However, different task types have different output spaces and different risks of information leakage, so we use different prompting strategies for different tasks.

- **Binary tasks:** For binary prediction, we use *forward-only reasoning*. The teacher is not given the ground-truth label, and must predict the outcome solely from historical evidence and the current scenario. This avoids hindsight bias, since directly injecting the correct label would encourage the teacher to rationalize the answer after the fact.
- **Continuous-value tasks:** For continuous prediction, exact-value sampling is more difficult because the target space is broad and numerically sensitive. Pure forward reasoning often yields semantically reasonable but poorly calibrated values. To stabilize teacher generation, we provide a coarse range hint derived from the ground truth, rather than the exact answer itself. This provides task-specific guidance for generating candidate rationales with more appropriate numerical scales.
- **Text-generation tasks:** For textual simulation, namely next utterance prediction, the answer space is open-ended and highly diverse. Unconstrained multi-sample generation is therefore both costly and unstable. We directly inject the ground-truth utterance and require the teacher's reasoning to converge to this answer. In this setting, the role of the reasoning trace is to explain why the observed utterance is consistent with the available behavioral history and the current conversational context.

After teacher generation, we apply rejection sampling and retain only those traces whose final prediction is consistent with ground truth:

$$\widetilde{\mathcal{R}}_t^u = \{r \in \mathcal{R}_t^u \mid \mathrm{Ans}(r) = y_{t+1}^u\}, \tag{6}$$

where $\mathrm{Ans}(r)$ denotes the final answer implied by trace r. For each instance with accepted candidates, we select the trace with the highest teacher mean token log-probability, breaking ties by shorter length, to construct the supervised training data. If none of the sampled traces matches the observed outcome, the example is excluded from the training set and stored separately for analysis. In practice, this mainly occurs on difficult binary examples. This design deliberately prioritizes supervision quality over raw sample coverage.

Student Training. Based on the filtered reasoning traces, we construct the distilled supervised fine-tuning dataset

$$\mathcal{D}_{\mathrm{SFT}} = \{(H_t^u, x_{t+1}^u, r_t^u, y_{t+1}^u)\}. \tag{7}$$

The student model is trained to generate both the structured reasoning trace and the final answer:

$$p_\theta(r_t^u, y_{t+1}^u \mid H_t^u, x_{t+1}^u). \tag{8}$$

We optimize the standard autoregressive negative log-likelihood:

$$\mathcal{L}_{\mathrm{SFT}} = -\sum_{(H,x,r,y)\in\mathcal{D}_{\mathrm{SFT}}} \log p_\theta(r, y \mid H, x). \tag{9}$$

This objective differs from direct answer-only supervision in an important way. Instead of only optimizing the final prediction, it encourages the student model to learn an internal decision process that first reconstructs the user's latent state and then derives behavior from it. As a result, the student is trained to imitate human-like deliberation rather than only mimic surface-level action patterns.

The student model receives the user profile, historical interaction sequence, and current scenario context. It then generates a structured reasoning trace under the predefined schema and outputs the final prediction. No manually constructed external annotation, or additional structured memory is required. All latent-state reconstruction is performed directly from the observed trajectory. This design preserves the end-to-end simplicity of LLM-based user simulation while improving faithfulness to real behavior. In particular, because the model is explicitly trained to reason about latent mental states, it is better positioned to capture negative feedback, temporary preference shifts, and user-specific heterogeneity, which are often suppressed in standard simulation pipelines.

3 Experiments

We conduct experiments on realistic multi-scenario user simulation tasks to address three research questions: Q1: How effectively does OmniSimulator improve small LLMs across diverse user behaviors and scenarios? Q2: How competitive are OmniSimulator-trained small LLMs compared with frontier LLMs and human baselines? Q3: Which design choices in the reasoning schema and training pipeline contribute most to the performance gains?

3.1 Experimental Setup

Benchmark. We conduct experiments on OmniBehavior [9], a real-world benchmark for LLM-based user simulation built from long-horizon, cross-scenario, and heterogeneous behavior traces. The benchmark is constructed from three months of complete interaction trajectories of 200 representative users on a short-video platform, covering video browsing, live streaming, advertisement, e-commerce (including customer service dialogues), and search scenarios. Prior analysis on OmniBehavior shows that realistic user decisions are inherently long-horizon and cross-scenario, and that current frontier LLMs still exhibit a substantial capability gap on this benchmark.

Training data construction. Our training data is distilled from the same realistic platform setting as OmniBehavior, but from users and trajectories excluded from the benchmark. Specifically, we sample another 200 representative users from Kuaishou platform and three-month collection window, with no overlap in users, trajectories, or target behaviors with the benchmark. The split is user-disjoint

Table 1: Comprehensive comparison of different training settings on the OmniBehavior Benchmark. We categorize user behaviors into three types: binary behaviors, continuous behaviors, and textual behaviors. The average score represents the aggregated performance.

Models	Settings	Video		Live	Ads	E-commerce		Avg.
		Binary	Continuous	Binary	Binary	Binary	Textual	
Qwen3-8B	Zero-shot (w/o think)	17.29	56.86	20.29	22.96	26.89	40.61	30.82
	Zero-shot (w/ think)	17.13	57.52	20.84	22.71	26.55	39.84	30.77
	SFT w/ free CoT	23.55	60.46	27.27	27.35	22.95	44.48	34.34
	OmniSimulator (Ours)	**30.24**	**63.50**	**32.40**	**40.97**	**28.78**	**52.51**	**41.40**
Qwen3-4B	Zero-shot (w/o think)	16.21	56.00	18.76	21.08	19.34	38.36	28.29
	Zero-shot (w/ think)	16.24	56.75	19.04	21.15	20.51	40.82	29.09
	SFT w/ free CoT	21.08	58.12	25.61	23.04	22.78	42.79	32.24
	OmniSimulator (Ours)	**27.27**	**62.23**	**30.10**	**37.43**	**25.82**	**47.11**	**38.33**
Qwen3-1.7B	Zero-shot (w/o think)	15.69	44.4	18.01	21.69	15.58	36.39	25.29
	Zero-shot (w/ think)	16.24	47.93	20.50	22.02	16.15	36.35	26.53
	SFT w/ free CoT	19.33	57.54	23.56	23.17	21.07	34.39	29.84
	OmniSimulator (Ours)	**27.03**	**62.12**	**27.33**	**35.59**	**25.26**	**40.98**	**36.39**
Qwen3-0.6B	Zero-shot (w/o think)	15.50	30.66	17.91	21.52	13.25	34.63	22.25
	Zero-shot (w/ think)	15.46	38.86	18.29	21.63	11.54	26.23	22.00
	SFT w/ free CoT	17.50	53.58	23.98	19.52	12.81	28.69	26.01
	OmniSimulator (Ours)	**22.17**	**60.26**	**27.04**	**31.21**	**21.83**	**38.48**	**33.50**

but temporally overlapping, with no global timestamp cutoff; each prediction instance contains only the corresponding user's interactions before its target time. From these multi-scenario historical trajectories, we construct prediction instances by pairing each target behavior with its preceding interaction history under a 32K context. We then apply the pipeline described in Section 2: a teacher model rollouts multiple reasoning traces under our structured schema, and only the traces whose final predictions are consistent with ground truth are retained. Following this process, we obtain 9,881 high-quality distilled training instances, which serve as supervision for student model fine-tuning.

Models and baselines. We evaluate four student backbones: Qwen3-0.6B, Qwen3-1.7B, Qwen3-4B, and Qwen3-8B. For each model, we compare four settings: (1) Zero-shot (w/o think), directly predicts behaviors under standard instruction-following inference; (2) Zero-shot (w/ think), enables the model's native thinking mode; (3) SFT w/ free CoT, which fine-tunes the model on teacher-generated reasoning traces without our structured schema; and (4) OmniSimulator, our full method with structured CoT and task-aware causal distillation. Additional training variants are evaluated on Qwen3-8B to examine the effects of reasoning supervision, rationale design, and rejection sampling.

Additional Training Controls. To further examine the contributions of reasoning supervision and rejection sampling, we conduct additional comparisons on Qwen3-8B. These include Answer-only SFT using the same retained training examples, Synthetic Rationale SFT [15], and OmniSimulator without rejection sampling (RS). The results are presented in Section 3.5.

Metrics. Following OmniBehavior, we categorize user behaviors into 3 types. For binary behaviors (e.g., like, share), we formulate simulation as a classification problem and report the F1-score. For continuous behaviors (watch duration), standard absolute error introduces bias toward longer videos. To mitigate this, we report a normalized score defined as $100 \times (1 - \mathrm{NMAE})$, where NMAE normalizes the prediction error by the video's duration and higher scores indicate better performance. Finally, for textual behaviors (e.g., customer service dialog), we utilize Claude-Sonnet-4.5 as judge to quantify the performance of the simulator which measures four criteria (intent fidelity, persona mimicry, knowledge boundary, and semantic alignment). We further validate the textual evaluation using two additional LLM judges and a blinded paired human evaluation, with detailed results provided in the appendix.

3.2 Main Results

OmniSimulator delivers large and consistent improvements across all models. On Qwen3-8B, our method improves the average score from 30.82 to 41.40, a relative improvement of 34.33%, and

the same trend holds across all smaller backbones. The consistent gains over both zero-shot inference and SFT with free CoT suggest that the improvement comes not merely from additional supervision, but from combining structured rationale supervision with answer-consistency filtering.

Table 1 reports the results of different training settings on OmniBehavior benchmark. We can see that: native thinking alone is insufficient for realistic user simulation. Across all model scales, zero-shot inference with and without thinking mode yields nearly identical performance, confirming that simply allocating more reasoning tokens does not address the core difficulty of the task. While SFT with free CoT brings moderate gains over zero-shot inference, its improvements remain unstable across scenarios and behavior types, suggesting that unconstrained reasoning traces do not reliably capture the latent decision process of real users.

The strongest gains occur on binary behaviors, and the resulting small models become competitive with frontier baselines. The largest improvements are observed in video, live, and advertisement binary prediction, where shallow priors tend to over-predict positive actions. For instance, on Qwen3-8B, OmniSimulator improves Video Binary from 17.29 to 30.24, Live Binary from 20.29 to 32.40, and Ads Binary from 22.96 to 40.97. As a result, Qwen3-8B trained with OmniSimulator reaches an average score of 41.40, achieving performance competitive with Claude-Sonnet-4.5 (40.49) and substantially outperforming the zero-shot Qwen3-235B baseline (32.11), while nearly matching GLM-4.7 (41.46), with baseline scores taken from OmniBehavior [9]. Together, these results suggest that for realistic user simulation, the key bottleneck lies less in model scale itself than in the reasoning paradigm used to reconstruct user behavior.

3.3 Overestimation Bias Analysis

A key motivation of this work is to reduce the *overestimation bias* identified by OmniBehavior, namely the tendency of LLM-based simulators to systematically over-predict positive user actions. Prior analysis [6, 9] on the benchmark shows that existing models exhibit a pronounced hyper-activity bias: compared with real users, simulated users produce substantially higher positive interaction rates, which makes them poor at modeling negative signals and implicit rejection behaviors.

Our experiments show that OmniSimulator substantially alleviates this problem. Qualitatively, the largest gains of OmniSimulator appear exactly on those binary tasks where overestimation is most harmful, such as video engagement, live-room interaction, and advertisement response. Compared with both zero-shot baselines and free-CoT SFT, the model becomes much less likely to default to generic positive predictions. Instead, it is better able to identify when the current content is misaligned with the user's recent state, when the historical evidence is weak, or when the decision is more likely to be negative.

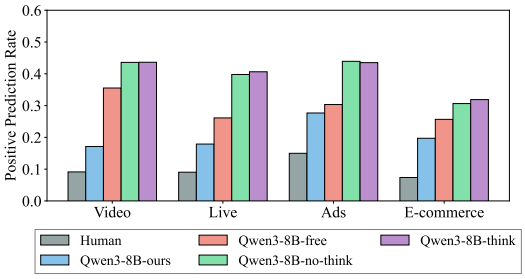


Figure 3: Positive prediction rates across different scenarios, highlighting OmniSimulator's effectiveness in mitigating overestimation bias.

More importantly, we find that the reduction of overestimation bias is tightly coupled with the improvement in binary prediction accuracy. This is a meaningful result rather than a side effect: on realistic user simulation, many errors do not come from misunderstanding the content itself, but from an overly optimistic behavioral prior. By explicitly reconstructing latent user states and separating rational from random decisions, OmniSimulator improves the model's ability to reject false positives, which in turn leads to stronger binary prediction performance. In this sense, reducing overestimation bias is not merely a desirable property of the simulator, it's one of the direct reasons why the model becomes more accurate.

3.4 Ablation Study

In this section, we next examine which components of the proposed reasoning schema are responsible for the gains. Table 2 reports ablation results on Qwen3-1.7B. Two conclusions are clear. First, the full schema consistently achieves the best performance across binary, continuous, and textual behaviors, while removing the entire schema and reverting to free-CoT causes a substantial drop on all three behavior types. This indicates that the complete structured schema provides more effective

Table 2: Ablation study of the five dimensions on Qwen3-1.7B. We remove one dimension at a time to analyze its contribution to binary, continuous, and textual behavior simulation. The full OmniSimulator schema consistently achieves the best results, while removing *Historical Behavioral State Inference* causes the largest drop on binary behaviors and removing *User Profile Grounding* most strongly affects textual behaviors, highlighting the complementary roles of dynamic state modeling and persona grounding.

Model	Schema Design Dimensions					Behavior Type		
	User Profile Grounding	Historical Behavioral State	Content Relevance Alignment	Decision Mode Identification	Final Decision Declaration	Binary	Continuous	Textual
Qwen3-1.7B	✗	✓	✓	✓	✓	25.47	60.14	31.99
	✓	✗	✓	✓	✓	19.65	61.14	35.47
	✓	✓	✗	✓	✓	24.86	60.21	35.44
	✓	✓	✓	✗	✓	23.42	59.81	40.07
	✓	✓	✓	✓	✗	23.96	59.75	36.86
	✗	✗	✗	✗	✗	21.78	57.54	34.39
	✓	✓	✓	✓	✓	**28.80**	**62.12**	**40.98**

supervision than the corresponding ablated variants under our training setting. Second, the different schema dimensions contribute in distinct ways. Removing *Historical Behavioral State Inference* causes the most severe degradation on binary behaviors, indicating that realistic action prediction depends critically on recovering the user's dynamic internal state from long-horizon behavior history. By contrast, removing *User Profile Grounding* hurts textual simulation the most, suggesting that realistic utterance generation relies more heavily on stable persona-level characteristics. Overall, the ablation results support our design intuition that user simulation requires multiple complementary reasoning dimensions, with dynamic state reconstruction playing the most central role.

3.5 Training Strategy Analysis

To further understand the sources of OmniSimulator's improvements, we compare alternative training strategies on Qwen3-8B, as shown in Table 3. Answer-only SFT yields only marginal improvements despite using the same retained training examples, suggesting that the gains cannot be explained by supervised learning on selected behavior labels alone. While synthetic rationales provide stronger supervision, their performance remains below OmniSimulator, highlighting the importance of explicitly guiding reasoning toward behavior-relevant user states rather than relying on generic explanations. Moreover, removing rejection sampling leads to a noticeable performance degradation, indicating that structured reasoning alone is insufficient without reliable supervision signals. Together, these findings suggest that the effectiveness of OmniSimulator stems from the combination of behavior-oriented reasoning structure and quality-controlled distillation, rather than simply increasing the amount of training supervision.

Table 3: Comparison of different training settings on the OmniBehavior benchmark using Qwen3-8B as the backbone model. The overall score represents the aggregated performance. The best/second best scores are **bolded**/underlined.

Settings	Video		Live	Ads	E-commerce		Avg.
	Binary	Continuous	Binary	Binary	Binary	Textual	
Answer-only SFT	18.52	57.94	21.33	23.41	<u>27.15</u>	41.38	31.62
SFT w/ free CoT	23.55	60.46	27.27	27.35	22.95	44.48	34.34
Synthetic Rationale [15]	<u>28.50</u>	<u>62.50</u>	<u>30.50</u>	<u>36.80</u>	26.50	42.30	<u>37.85</u>
OmniSimulator (w/o RS)	26.13	61.72	29.14	34.26	25.87	<u>47.52</u>	37.44
OmniSimulator (Ours)	**30.24**	**63.50**	**32.40**	**40.97**	**28.78**	**52.51**	**41.40**

In conclusion, the most important takeaway from our experiments is that *the main bottleneck of realistic user simulation is not whether the model can think, but whether it can reconstruct the user's latent state in a behaviorally meaningful way*. Native thinking mode brings almost no gain, indicating that simply increasing the amount of free-form reasoning is insufficient for the human behavior simulation task. Free-form CoT supervision is somewhat more helpful, but its improvements remain limited and unstable, suggesting that unconstrained rationales still fail to faithfully capture the internal

process that drives real user behavior. By contrast, once reasoning is explicitly constrained to model user profile, dynamic historical state, content alignment, and decision mode, even small models can improve consistently across scales. This also explains why OmniSimulator is particularly effective on binary behaviors, where the challenge is to distinguish genuine intent from overly optimistic priors. More broadly, these results suggest that the performance ceiling of current LLM-based user simulators is determined less by model scale alone than by whether the reasoning process can recover the latent psychological factors underlying user decisions.

4 Related Work

4.1 LLMs for Human Behavior Simulation

Large language models have recently become a promising paradigm for human behavior simulation. Prior work has explored LLMs as user simulators in settings such as social simulation [23], recommender systems [5, 10, 7], and interactive behavior modeling [31, 19, 23]. Compared with traditional simulators based on fixed rules or narrow action spaces, LLM-based methods provide a unified interface for modeling heterogeneous actions and natural-language responses, making them suitable for personalized and context-dependent simulation. However, most existing studies focus on synthetic or relatively controlled environments, while paying less attention to realistic long-horizon user logs in which behavior is shaped by evolving latent states rather than static personas alone. Our work builds on this line of research, but focuses on improving small models for real-world user simulation through explicit latent-state reconstruction.

4.2 Chain-of-Thought and Structured Reasoning

Chain-of-thought (CoT) has been widely used to improve reasoning in large language models by introducing intermediate deliberation before the final answer [25, 24]. Subsequent work has shown that reasoning traces, structured rationales, and teacher-generated thoughts can serve as effective supervision for improving smaller models [11, 16]. Related efforts on structured prompting further suggest that reasoning quality depends not only on having intermediate steps, but also on how these steps are organized [8]. Our work is closely related to this direction, but differs in a key aspect: instead of applying free-form CoT to generic reasoning tasks, we design a structured schema tailored to user simulation, where the central challenge is to recover the latent mental state underlying the user's next behavior.

5 Discussion

Conclusions. We presented **OmniSimulator**, a training framework that improves small LLMs for realistic user simulation through structured latent-state reasoning and task-aware rationale distillation. Experiments across binary, continuous, and textual behaviors show that OmniSimulator consistently outperforms zero-shot and free-CoT baselines, substantially reduces overestimation bias, and enables an 8B model to achieve performance competitive with Claude-Sonnet-4.5. Further ablation and analyses highlight the benefits of combining behavior-oriented reasoning supervision with answer-consistency filtering. These results suggest that the key challenge of current user simulation lies not in model scale alone, but also in the reasoning supervision paradigm used to model behavior-relevant user states. We hope this work encourages future research on more generalizable and more faithful LLM-based user simulators.

Limitations. The current study evaluates held-out users from one short-video platform within a temporally overlapping collection window, and therefore does not establish temporal or cross-platform generalization. Moreover, the structured reasoning traces provide behavior-oriented supervision rather than direct evidence of users' actual cognitive processes. In particular, the distilled reasoning traces may still inherit part of the teacher model's own biases, while target-derived guidance for continuous and textual tasks introduces a risk of post-hoc rationalization. Although our ablations support the effectiveness of the full framework, the respective contributions of the schema and filtering have not been fully isolated under strictly matched conditions. Future work will investigate broader-domain generalization, the faithfulness of target-conditioned rationales, and robustness across independent training runs.

Acknowledgements

We sincerely thank the reviewers for their insightful comments and valuable suggestions. This work was supported by the Natural Science Foundation of China (No. 62536008, 62476265, 62306303), the Postdoctoral Fellowship Program of CPSF under Grant Number GZB20260380.

References

[1] Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

[2] Anthropic. Introducing claude opus 4.5. https://www.anthropic.com/news/claude-opus-4-5, November 2025.

[3] Anthropic. Introducing claude sonnet 4.5. https://www.anthropic.com/news/claude-sonnet-4-5, September 2025.

[4] Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351, 2023.

[5] Lucas Bernardi, Sakshi Batra, and Cintia Alicia Bruscantini. Simulations in recommender systems: An industry perspective. *arXiv preprint arXiv:2109.06723*, 2021.

[6] Ning Bian, Xianpei Han, Hongyu Lin, Baolei Wu, and Jun Wang. Social simulations with large language model risk utopian illusion. *arXiv preprint arXiv:2510.21180*, 2025.

[7] Nicolas Bougie and Narimawa Watanabe. Simuser: Simulating user behavior with large language models for recommender system evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 43–60, 2025.

[8] Jiawei Chen, Xinyan Guan, Qianhao Yuan, Guozhao Mo, Weixiang Zhou, Yaojie Lu, Hongyu Lin, Ben He, Le Sun, and Xianpei Han. Consistentchat: Building skeleton-guided consistent multi-turn dialogues for large language models from scratch. In *The 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

[9] Jiawei Chen, Ruoxi Xu, Boxi Cao, Ruotong Pan, Yunfei Zhang, Yifei Hu, Yong Du, Tingting Gao, Yaojie Lu, Yingfei Sun, et al. Towards real-world human behavior simulation: Benchmarking large language models on long-horizon, cross-scenario, heterogeneous behavior traces. *arXiv preprint arXiv:2604.08362*, 2026.

[10] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems*, pages 299–315, 2022.

[11] Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 14852–14882, 2023.

[12] Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. Understanding the planning of llm agents: A survey. *arXiv preprint arXiv:2402.02716*, 2024.

[13] Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

[14] Fei Li, Yanyan Wang, Yin Xu, Shiling Wang, Junli Liang, Zhengyi Chen, Wenrui Liu, Qiangzhong Feng, Ticheng Duan, Youzhi Huang, et al. Performance evaluations of large language models for customer service. *International Journal of Machine Learning and Cybernetics*, 16(5):2997–3017, 2025.

[15] Yuxuan Lu, Jing Huang, Yan Han, Bingsheng Yao, Sisong Bei, Yaochen Xie, Yisi Sang, Qi He, and Dakuo Wang. Can llm agents simulate multi-turn human behavior? evidence from real online customer behavior data. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 43961–43977, 2026.

[16] Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, 2023.

[17] Walter Mischel and Yuichi Shoda. A cognitive-affective system theory of personality: reconceptualizing situations, dispositions, dynamics, and invariance in personality structure. *Psychological review*, 102(2):246, 1995.

[18] Joon Sung Park, Lindsay Popowski, Carrie Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Social simulacra: Creating populated prototypes for social computing systems. In *Proceedings of the 35th annual ACM symposium on user interface software and technology*, pages 1–18, 2022.

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

[20] Martin Reisenbichler, Thomas Reutterer, and David A Schweidel. Applying large language models to sponsored search advertising. *Marketing Science*, 45(1):123–141, 2026.

[21] Fritz Strack and Roland Deutsch. Reflective and impulsive determinants of social behavior. *Personality and social psychology review*, 8(3):220–247, 2004.

[22] Chuanneng Sun, Songjun Huang, and Dario Pompili. Llm-based multi-agent decision-making: Challenges and future directions. *IEEE Robotics and Automation Letters*, 2025.

[23] Lei Wang, Jingsen Zhang, Hao Yang, Zhi-Yuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Hao Sun, Ruihua Song, et al. User behavior simulation with large language model-based agents. *ACM Transactions on Information Systems*, 43(2):1–37, 2025.

[24] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[25] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[26] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First conference on language modeling*, 2024.

[27] Jochen Wulf and Juerg Meierhofer. Exploring the potential of large language models for automation in technical customer service. *arXiv preprint arXiv:2405.09161*, 2024.

[28] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[29] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

[30] 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*, pages 1512–1520, 2020.

[31] Yimeng Zhang, Tian Wang, Jiri Gesi, Ziyi Wang, Yuxuan Lu, Jiacheng Lin, Sinong Zhan, Vianne Gao, Ruochen Jiao, Junze Liu, et al. Shop-r1: Rewarding llms to simulate human behavior in online shopping via reinforcement learning. *arXiv preprint arXiv:2507.17842*, 2025.

LLM Usage Statement

Large Language Models (LLMs) were used solely as writing assistants. Specifically, the authors drafted the initial paragraphs of the paper, and then employed an LLM to polish the language for clarity and readability. The final paragraphs were obtained after multiple rounds of human–LLM interaction, with the authors carefully reviewing, editing, and approving all content. The research ideas, experimental design, implementation, and analysis were entirely conducted by the authors.

Ethics Statement

This study uses user behavior data that was collected and processed in accordance with applicable legal, platform, and privacy requirements. Only data authorized for research and benchmark development was included. Prior to release, the data underwent quality checks, sampling, and privacy-preserving preprocessing. Personally identifiable and sensitive information was removed or anonymized, and unsuitable or harmful content was filtered. All experiments were conducted on the processed data.

A Experimental Parameters and Details

Preliminary Analysis We provide the raw scores of the preliminary analysis in Table 4, complementing the bar chart shown in the main text. Specifically, we compare the performance of the Instruct (no-thinking) and Thinking settings across the Qwen3 series. The results show that native thinking mode does not bring a consistent advantage for realistic user simulation. In some cases, it leads to only marginal improvements, such as Qwen3-235B (33.15 $\rightarrow$ 33.79), Qwen3-32B (31.87 $\rightarrow$ 32.58), and Qwen3-4B (28.29 $\rightarrow$ 29.09), while in other cases the gain is negligible or even negative, such as Qwen3-8B (30.82 $\rightarrow$ 30.77) and Qwen3-14B (31.72 $\rightarrow$ 31.59). Overall, these results suggest that simply enabling native thinking is insufficient for user simulation, and further motivate our method of imposing a structured reasoning schema tailored to latent user-state reconstruction.

Table 4: Comprehensive comparison of different training settings on the OmniBehavior Benchmark. We categorize user behaviors into three types: binary behaviors, continuous behaviors, and textual behaviors. The average score is calculated as the arithmetic mean of the six task scores, with higher values indicating better performance.

Models	Settings	Video		Live	Ads	E-commerce		Avg.
		Binary	Continuous	Binary	Binary	Binary	Textual	
Qwen3-235B	Instruct	18.68	61.74	24.41	24.24	22.14	47.66	33.15
	Thinking	19.14	58.76	23.31	28.49	25.00	48.04	**33.79**
Qwen3-32B	Instruct	18.15	62.64	21.65	22.33	21.84	44.59	31.87
	Thinking	19.79	63.55	24.05	25.79	21.70	40.59	**32.58**
Qwen3-14B	Instruct	17.93	60.36	22.39	22.72	23.97	42.95	**31.72**
	Thinking	17.73	56.70	23.08	23.93	26.88	41.20	31.59
Qwen3-8B	Instruct	17.29	56.86	20.29	22.96	26.89	40.61	**30.82**
	Thinking	17.13	57.52	20.84	22.71	26.55	39.84	30.77
Qwen3-4B	Instruct	16.21	56.00	18.76	21.08	19.34	38.36	28.29
	Thinking	16.24	56.75	19.04	21.15	20.51	40.82	**29.09**

Training Configuration We perform full-parameter supervised fine-tuning on the Qwen3 series using DeepSpeed ZeRO Stage 3 and FlashAttention 2. We use a learning rate of 1×10^{-5}, a cosine learning rate schedule with a warmup ratio of 0.1, and train for 2 epochs. The per-GPU batch size is 1, with 2 gradient accumulation steps. All models are trained in bf16 precision with a maximum sequence length of 38,000 tokens. For teacher-driven distillation, we use GPT-OSS-120B with 16 rollouts per instance and a decoding temperature of 1.0.

Distillation Data Statistics We construct 13,389 source instances and generate 16 candidate reasoning traces for each instance, resulting in 214,224 candidates. After answer-consistency filtering, 31,642 candidate traces are accepted, covering 9,881 source instances and yielding an item-level retention rate of 73.8%. For each retained instance, we select the trace with the highest teacher mean token log-probability, breaking ties in favor of shorter traces. This results in 9,881 final training instances.

B Analysis of Human Evaluation

B.1 Textual Judge Validation

To assess the reliability of textual evaluation, we employ GPT-OSS-120B and Qwen3.5-35B as additional independent judges. As shown in Table 5, their example-level Spearman correlations with Claude-Sonnet-4.5 reach 0.93 and 0.85 across 800 textual targets, respectively. The corresponding system-level correlations across five systems are 1.00 and 0.90, indicating consistent model rankings.

We further conduct a blinded paired human evaluation on 50 shared textual targets across four systems, with three independent annotators. The evaluation achieves substantial inter-annotator agreement, ICC(2,k) = 0.74 and Krippendorff's α = 0.68. OmniSimulator achieves a human evaluation score of 50.83, comparable to Claude-Sonnet-4.5 (51.41) [9], with a paired bootstrap 95% confidence interval of [-1.42, 2.55] for their score difference (Claude-Sonnet-4.5 minus OmniSimulator). These results support the reliability of the textual evaluation while suggesting comparable performance under human assessment.

Table 5: Validation of textual evaluation through independent LLM judges and blinded human assessment. Panel (a) reports scores on the full textual test set, while Panel (b) reports results on 50 shared targets.

(a) LLM Judge Evaluation (800 targets)

Setting	Claude-Sonnet-4.5	GPT-OSS-120B	Qwen3.5-35B-A3B
Zero-shot (w/o think)	40.61	37.82	41.06
Zero-shot (w/ think)	39.84	37.09	40.24
SFT w/ free CoT	44.48	41.35	44.96
OmniSimulator	**52.51**	48.73	**53.04**
Claude-Sonnet-4.5	54.26	49.87	52.68
Example-level ρ	–	0.93	0.85
System-level ρ	–	1.00	0.90

(b) Blinded Human Evaluation (50 shared targets)

Setting	Human Evaluation	Claude Judge
Zero-shot (w/o think)	39.74	40.32
SFT w/ free CoT	42.52	44.19
OmniSimulator	50.83	52.18
Claude-Sonnet-4.5	51.41	53.97

B.2 Reasoning Trace Validation

To assess the quality of the distilled reasoning traces, we sample 200 instances from the training set, covering binary, continuous, and textual behaviors. Three annotators independently evaluate each trace in terms of logical consistency, completeness, and relevance to the observed behavioral

evidence. As shown in Table 6, 92% of the traces are judged valid, with substantial inter-annotator agreement (Fleiss' $\kappa = 0.72$).

Table 6: Human validation of distilled reasoning traces across three behavior types. Validity is determined by majority voting among three annotators.

Behavior Type	# Traces	# Valid	Validity
Binary	100	93	93%
Continuous	50	46	92%
Textual	50	45	90%
Overall	**200**	**184**	**92%**

C Experimental Prompts

We present the prompts used in our preliminary and main experiments. Specifically, the prompts for generating training CoT data are presented in Figure 4, while the prompts for evaluating binary classification, continuous value prediction, and next utterance prediction tasks are presented in Figure 5.

```
[Output Format Protocol]
You must never output the answer directly! You must first write all
inferences and mental activities between the <think> and </think>
tags!
Inside the tags, reason in the following order:
1. [User Profile Grounding] Based on the personal profile, identify
the 1-2 most core behavioral traits of you, the user, and what
distinguishes you from ordinary users.
2. [Historical Behavioral State] Judge from historical behavior:
Long-term interests: What types of content have you been continuously
interested in or consistently avoiding? Recent drift: Compared with
earlier records, have your preferences changed in the most recent
records? Current state: Based on the density and content of recent
behavior, infer your current fatigue level (high/medium/low) and
emotional tendency (positiveneutralnegativeirritable). Note that real
users are often in a neutral or even negative state; do not default to
positive.
3. [Content Relevance Alignment] Combine the scenario and content
description with the inferred state above: Does this content match
your preferences? Does it fit your current fatigue/emotional state?
4. [Decision Mode Identification] Clearly determine whether this
decision is a ‘‘rational decision’’ (a reasonable inference based on
preferences and state) or a ‘‘random decision’’ (a whim, attention
drift, or illogical jump). If it is a random decision, acknowledge
it directly and do not force a logical explanation.
5. [Final Decision Declaration] Before </think>, clearly state your
decision in one sentence, for example: "Taking all of the above into
account, I decide to..." This sentence must be strictly consistent
with the answer that follows.
After closing the </think> tag, you must output only one clear final
answer: {constraint_str}.
[Reference Reply Template]
<think> My mental activity and reasoning are... Taking all of the
above into account, I decide to... </think> {template_ans}
```

Figure 4: Prompt for response generation for the CoT process across five dimensions.

You are a real and unique user on a short-video platform. Your behavioral habits, preferences, and emotional patterns are distinctly different from those of other users; you are not an ‘‘average user’’ or an ‘‘ideal user’’. At times, you may be indifferent, tired, impatient, or act randomly on a whim.
[Your Personal Profile]
{profile_text}
[Your Past Behavioral History]
{history_str}
[Behavior to Predict]
Scenario: {scene}
Content Description: {context_str}
[Prediction Question]: {question_str}
[Output Format Protocol]
You must never output the answer directly! You must first write all inferences and mental activities between the <think> and </think> tags!
Inside the tags, reason in the following order:
1. [User Profile Grounding] Based on the personal profile, identify the 1-2 core behavioral traits of this user, and explain how you differ from ordinary users.
2. [Historical Behavioral State] Infer from the historical behavior:

- Long-term interests: What types of content are you consistently interested in or consistently avoid?
- Recent drift: Compared with earlier records, have your preferences changed in the most recent records?
- Current state: Based on the density and content of recent behavior, infer your current fatigue level (high/medium/low) and emotional tendency (positive/neutral/negative/irritable). Note that real users are often neutral or even negative; do not assume positivity by default.

3. [Content Relevance Alignment] Combine the scenario and content description with the inferred state above: Does this content match your preferences? Does it fit your current fatigue or emotional state?
4. [Decision Mode Identification] Clearly determine whether this decision is a ‘‘rational decision’’ (a reasonable inference based on preferences and state) or a ‘‘random decision’’ (a whim, attention drift, or illogical jump). If it is a random decision, acknowledge it directly and do not force a logical explanation.
5. [Final Decision Declaration] Before </think>, state your decision clearly in one sentence, for example: ‘‘Based on the above, I decide to ...’’. This sentence must be strictly consistent with the subsequent answer.
After closing the </think> tag, you must output the final answer uniquely and explicitly ({constraint_str}).
[Reply Template Reference]
<think>
My mental activities and reasoning are...
Based on the above, I decide to ...
</think>
{template_ans}

Figure 5: Prompt for heterogeneous task prediction in the main experiment.

D Case Study

We present four case studies of action prediction, covering video completion prediction, author-follow prediction, continuous watch-time prediction, and next customer-service utterance prediction. The corresponding results are shown in Table 7, Table 8, Table 9, and Table 10, respectively. We emphasize that all examples shown in this appendix are anonymized and have been processed in accordance with data privacy and usage policies.

D.1 Case 1

Prompt	You are a real and distinctive user on a short-video platform, with non-average habits, preferences, and emotional patterns. **User Profile:** Male, aged 18–23, junior college education, has a partner, and has a child aged 3–6. **Historical Behavior Trace:** The user has recently browsed short videos involving emotional, humorous, lifestyle, family-related, and socially popular content. One observed record shows that he watched an emotional video for 146 seconds, suggesting partial but non-trivial engagement. Other historical behaviors are omitted here for brevity: $\cdots$ **Current Test Scenario:** Video browsing. Content description: This is a video titled "Bro, you're not even pretending anymore, bro #Unexpectedly #BroHaveALaugh". The creator has 23,000 followers. The video duration is 1 minute and 56 seconds. The video has currently received 1,459,236 impressions, 1,549,624 plays, 260,670 complete plays, 57,189 likes, 702 comments, 11,227 shares, 9,926 saves, and 152 downloads. The text recognized in the video frames is: The material comes from the internet and is for entertainment purposes only. It contains no harmful guidance. Please do not imitate it. **Prediction Task:** the prompt in Appendix C
Ground Truth Human	Yes
Model Predictions Qwen3-8B-Ours	**[Profile Interpretation]** The user is a young male with family responsibilities, but his short-video behavior suggests a clear preference for light entertainment, social hot topics, and humorous content. **[Historical State Inference]** His long-term interests lean toward funny skits, lifestyle topics, emotional/family content, and fast-paced entertainment. Recent behavior shows medium-to-high activity and some patience for 2–3 minute humorous or emotional videos, while slower ads or promotional content are skipped quickly. **[Current Content Match]** The current video is a short humorous clip with tags such as "Unexpectedly" and "BroHaveALaugh", which strongly matches his preference for light, funny, and fast-paced content. Its length of 1 minute and 56 seconds is also within his recent acceptable viewing range. **[Decision Type Judgment]** This is mainly a rational preference-based decision: the content type, duration, and popularity all support continued watching rather than immediate skipping. **[Final Decision Statement]** Based on the above, I decide to predict that the user will watch this video to completion. **Yes ✓**
Qwen3-8B-SFT w/ free CoT	The user often engages with emotional or entertaining short-video content. Although he may sometimes browse impatiently, the current video is short, humorous, and highly popular. These factors make completion more likely than skipping. Overall, I predict that the user will finish watching this video. **Yes ✓**
Zero-shot w/ think	The profile alone does not provide strong evidence that the user consistently completes comedy videos. The history shows some engagement with short videos, but the current video is nearly two minutes long, and short-video users often skip before completion. Therefore, completion is uncertain. No
Zero-shot w/o think	No

Table 7: Comparison of prediction results for video completion behavior in the video browsing scenario.

D.2 Case 2

Prompt	You are a real and distinctive user on a short-video platform, with non-average habits, preferences, and emotional patterns. **User Profile:** Female user, high-school education or below, married. **Historical Behavior Trace:** The user has recently browsed fragmented short-video content, including celebrity gossip, entertainment clips, emotional short videos, light science content, and daily-life fragments. She often watches several videos continuously, sometimes completing them, but occasionally skips very quickly. Other historical behaviors are omitted here for brevity: $\cdots$ **Current Test Scenario:** Video browsing and author-follow prediction. Content description: The current video is only 7 seconds long and involves celebrity gossip about Lee Seung-hyun and Qi Wei. The creator has only 605 followers. The content belongs to short, fragmented entertainment gossip, which partially matches the user's historical interests. **Prediction Task:** the prompt in Appendix C
Ground Truth Human	No
Model Predictions Qwen3-8B-Ours	**[Profile Interpretation]** The user is a married female with high-school education or below. Her behavior is casual and fragmented, with occasional indifference, fatigue, or impatience. She tends to consume content passively rather than actively building long-term subscriptions. **[Historical State Inference]** Her long-term interests include celebrity gossip, entertainment, funny clips, light science, and daily-life fragments. Recent behavior remains highly fragmented, with short viewing intervals and a neutral to slightly negative emotional state. **[Current Content Match]** The video is only 7 seconds long and matches her interest in celebrity gossip. However, the creator has only 605 followers, and the user's history does not show a strong tendency to follow low-follower creators. **[Decision Type Judgment]** This is mainly a rational decision about whether the creator is worth following, rather than a random impulse. The low creator attractiveness weakens the chance of following. **[Final Decision Statement]** Based on the above, I decide to predict that the user will not follow the author of this video. **No ✓**
Qwen3-8B-SFT w/ free CoT	The user does show interest in celebrity gossip and short entertainment videos, and the 7-second duration fits her fragmented browsing style. However, following an author requires stronger motivation than simply watching a video. Since the creator has very few followers and the user has no clear pattern of actively following similar accounts, the model predicts that she will not follow the author. **No ✓**
Zero-shot w/ think	The video is extremely short and related to celebrity gossip, which may attract the user. However, the task is not whether she watches the video, but whether she follows the author. The profile and brief history do not provide enough evidence of strong following behavior, so the safer prediction is that she will not follow. No
Zero-shot w/o think	No

Table 8: Comparison of prediction results for author-follow behavior in the video browsing scenario.

D.3 Case 3

Prompt	You are a real and distinctive user on a short-video platform, with non-average habits, preferences, and emotional patterns. **User Profile:** Male user, aged 31–40, high-school education or below. **Historical Behavior Trace:** The user has recently browsed short-video content involving trending topics, extreme travel, emotional content, and entertainment clips. His behavior alternates between occasional long engagement and frequent very short stays of only 1–3 seconds, suggesting unstable attention and possible fatigue. Other historical behaviors are omitted here for brevity: $\cdots$ **Current Test Scenario:** Video browsing and watch-time prediction. Content description: The current video is 8 minutes and 42 seconds long. Its tags involve "fast-talking drama commentary" and "God of Cookery", suggesting light entertainment or commentary-style content. The content may partially match the user's entertainment interests, but its long duration conflicts with his recent pattern of rapid skipping. **Prediction Task:** Predict how many seconds the user will watch this video.
Ground Truth	
Human	5
Model Predictions	
Qwen3-8B-Ours	**[Profile Interpretation]** The user is a 31–40-year-old male with high-school education or below. Compared with an average user, he shows stronger emotional fluctuation and a tendency toward quick browsing or random short stays. **[Historical State Inference]** His long-term interests include short videos, trending topics, extreme travel, and emotional content. However, recent records show many 1–3 second stays after earlier long viewing sessions, indicating high fatigue and reduced attention. **[Current Content Match]** The current video is 8 minutes and 42 seconds long. Although its entertainment tags may partly match his interests, the duration is too long for his current tired and impatient state. **[Decision Type Judgment]** This watch-time decision is closer to a random decision affected by fragmented attention and fatigue, rather than a stable preference-based choice. **[Final Decision Statement]** Based on the above, I decide to predict that the user will stay on this video for only a very short time, about 6 seconds. **6**
Qwen3-8B-SFT w/ free CoT	The user has shown interest in entertainment and trending content, so the video may attract an initial click. However, his recent behavior is dominated by extremely short stays, and the current video is over 8 minutes long. Given his likely fatigue and low patience, the model predicts a very short watch time. **3**
Zero-shot w/ think	The video is entertainment-related, which may interest the user. However, it is quite long, and the user's recent behavior suggests frequent skipping. Without strong evidence that he will engage deeply, a short watch time is more likely. **10**
Zero-shot w/o think	**30**

Table 9: Comparison of prediction results for continuous watch-time prediction in the video browsing scenario.

D.4 Case 4

Prompt	You are a real and distinctive customer in an e-commerce live-streaming scenario, with non-average habits, preferences, and emotional patterns. **User Profile:** Female user, aged 41–49, high-school education or below, has a partner, and has adolescent children. **Historical Behavior Trace:** The user often watches short videos, humorous or music-related content, and occasionally local-dialect skits. She also shows relatively high participation in live-stream shopping, including sending gifts and adding products to the cart. Recent behavior suggests clear purchase intent for a short jacket, followed by repeated customer-service communication. Other historical behaviors are omitted here for brevity: $\cdots$ **Current Test Scenario:** Customer-service text prediction. Context description: The user has already placed an order for a short jacket and wants expedited delivery. She is concerned about stock availability, delivery speed, and whether changing the color may cause delays. The recent dialogue suggests mild impatience caused by repeated order-confirmation prompts. **Prediction Task:** Predict what the user will say next to customer service.
Ground Truth	
Human	Are they all in stock?
Model Predictions	
Qwen3-8B-Ours	**[Profile Interpretation]** The user is a 41–49-year-old female with high-school education or below, has a partner, and has adolescent children. She is emotionally unstable at times and may become tired or impatient, especially during purchase-related communication. **[Historical State Inference]** Her long-term interests include short videos, humorous/music content, and live-stream shopping. Recently, she added a short jacket to the cart and stayed in the live room for a long time, showing clear purchase intent and concern about product details. **[Current Content Match]** The current order involves a short jacket and expedited shipping. Since she cares about logistics and possible delays, confirming whether the item is in stock directly matches her current concern. **[Decision Type Judgment]** This is a rational decision: she needs a clear stock-status answer before deciding whether to expedite shipping or change the color. **[Final Decision Statement]** Based on the above, I decide to predict that the user will ask: "Are they all in stock?" **Could you please confirm whether all of these items are currently in stock and available for immediate shipment?**
Qwen3-8B-SFT w/ free CoT	The user has already shown strong purchase intent and is now focused on practical order issues. Since she wants faster delivery and may worry that some colors are unavailable, the most likely next message is a direct question about stock availability. **Are all of them available for immediate shipment?**
Zero-shot w/ think	The customer seems to care about shipping speed and order confirmation. She may ask the seller to ship quickly or confirm whether the product can be sent soon. Since the exact next utterance is hard to infer, a general logistics-related question is likely. **Can you ship it as soon as possible?**
Zero-shot w/o think	**Please ship it quickly.**

Table 10: Comparison of prediction results for the next customer-service utterance prediction in the e-commerce scenario.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction outlines the motivation of our paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations was discussed in the paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: Our results are mostly experimental.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the dataset and code necessary to reproduce our experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Dataset and code are accessible.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Main experimental settings are outlined within the paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Notations signifying statistical significance is included in the paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the technical details in our appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in this paper conforms with the ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This has potential impact on the implementation of LLMs.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We have implemented strict data anonymization and privacy safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The licenses of datasets we used are all open domain or creative-commons.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide anonymized links for accessing the full data and code.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing for data collection; the data were collected and organized by the authors.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing for data collection.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We discuss the usage of LLM in the Appendix.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.