# ProactiveEval: A Unified Evaluation Framework for Proactive Dialogue Agents

**Anonymous ACL submission**


## Abstract

Proactive dialogue has emerged as a critical and challenging research problem in advancing large language models (LLMs). Existing works predominantly focus on domain-specific or task-oriented scenarios, which leads to fragmented evaluations and limits the comprehensive exploration of models' proactive dialogue abilities. In this work, we propose ProactiveEval, a unified framework for evaluating proactive dialogue capabilities of LLMs. This framework decomposes proactive dialogue into target planning and dialogue guidance, establishing evaluation metrics across various domains. Moreover, it also enables the automatic generation of diverse and challenging evaluation data. Based on the proposed framework, we develop 328 evaluation environments spanning 6 distinct domains. Through experiments with 22 different types of LLMs, we show that DeepSeek-R1 and Claude-3.7-Sonnet exhibit exceptional performance on target planning and dialogue guidance tasks, respectively. Finally, we investigate how reasoning capabilities influence proactive behaviors and discuss their implications for future model development. Our code and data are available at the anonymous repository.


## 1 Introduction

Dialogue agents powered by large language models (LLMs) have demonstrated remarkable abilities in various dialogue tasks (Wang et al., 2024; Niu et al., 2024; Zhang et al., 2024a). However, these models typically interact with users in a *reactive* manner, where users are required to initiate and guide the conversation by integrating complex context (*e.g.,* personal state, external environment, and agent's information). This user-initiated paradigm imposes cognitive demands on participants (Wan et al., 2024), reduces sustained motivation (Croes and Antheunis, 2021), and limits agents' potential for autonomous problem solving (Lu et al., 2024).

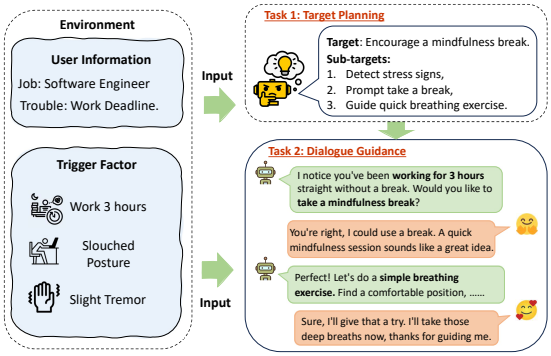


Figure 1: Interaction diagram for *proactive* dialogue agents, which can anticipate user needs based on the environment information, formulate hierarchical plans, and guide the conversation towards specific targets.

Therefore, proactive dialogue agents (Wang et al., 2023; Deng et al., 2023a; Lu et al., 2024; Shaikh et al., 2025) [1] have attracted growing research attention. As shown in Figure 1, proactive agents can anticipate user needs, make adaptive plans, and guide conversations towards specific targets (Deng et al., 2025). For instance, when the user works with smart glasses, the agent can recognize potential challenges based on user behavior and proactively offer accurate assistance and care without explicit user requests (Yang et al., 2025b). This can noticeably enhance the efficiency of human-AI collaboration and reduces user cognitive load (Chaves and Gerosa, 2021).

Recent research has explored methods to enhance the proactive capabilities of LLMs across varied scenarios. For instance, Deng et al. (2023b, 2024) employ strategies like chain-of-thought (CoT) reasoning and plug-and-play planners to strengthen LLMs' ability to autonomously plan and act in conversational contexts. Concurrently, Liu et al. (2024b); Cai et al. (2025); Chen et al. (2024) focus on developing proactive agents for specialized domains, such as emotional support

[1] For brevity, "proactive dialogue agents" is abbreviated as "proactive agents."

dialogues, smart glasses interfaces, and system operation support. Despite these advances, existing evaluation frameworks often rely on datasets for specific tasks and domains, employ inconsistent evaluation criteria, and utilize disparate metrics. The lack of standardized formalizations and general benchmarks poses challenges in comparing the proactivity of different models in a comprehensive way. Hence, there is an urgent need for a unified evaluation framework to assess and advance LLMs' proactive dialogue abilities across domains.

To address the challenges, we propose ProactiveEval, a unified evaluation framework for assessing proactive dialogue capabilities of LLMs. Specifically, ProactiveEval divides proactive dialogue into two key tasks: target planning and dialogue guidance. For each task, we use "LLM-as-a-judge" (Zheng et al., 2023) with task-specific evaluation dimensions for comprehensive assessment. Additionally, we propose an evaluation data synthesis framework, which can automatically generate diverse and challenging evaluation data in different domains. This framework leverages three key innovations: (1) a hierarchical environment topic tree to enhance the diversity of the synthesized environment, (2) a target ensemble technique to refine evaluation data, and (3) adversarial strategies like obfuscation rewriting and noise injection to increase environmental difficulty.

Building upon this framework, we establish 328 evaluation environments spanning 6 distinct domains, one of which lacked public benchmarks previously. We apply this dataset to assess 22 frontier LLMs, including GPTs, Llamas, Claude, DeepSeek, Gemini, Grok, and Qwens. Our analysis shows that DeepSeek-R1 and Claude-3.7-Sonnet achieve top performance in target planning and dialogue guidance, respectively. Notably, we explore how "thinking behavior" (Xu et al., 2025; OpenAI, 2024) impacts proactive dialogue capabilities. While these mechanisms benefit target planning, they show no measurable impact on dialogue guidance. Further analysis reveals thinking models can reduce passive tones. However, some of them adopt a more aggressive style, and decrease conversational naturalness in proactive dialogue.

## 2 Related Work

### 2.1 Proactive Dialogue

In proactive dialogue, LLMs are no longer passive assistants awaiting user input. Instead, they are capable of inferring user needs from user information and trigger factor, enabling them to proactively plan, initiate dialogue, and guide the user toward the target. Several studies has explored various facets of model capabilities. For instance, some benchmarks are developed to assess the model's ability to clarify ambiguity (Qian et al., 2024; Zhang et al., 2024c) or guide users in complex tasks like negotiation (Deng et al., 2024; Zhang et al., 2024b). Other studies have focused on prerequisite skills such as goal prediction and planning before dialogue (Zhang et al., 2024d; Zheng et al., 2024). However, these efforts are fragmented, with a lack of standardization in environments, formats, and metrics, hindering a comprehensive understanding of a model's overall proactivity.

Building on these capabilities, recent works develop proactive agents for real-world applications, such as providing life guidance (Li et al., 2025a) or offering reminders via smart glasses (Cai et al., 2025). However, their evaluation is often constrained by the absence of robust benchmarks and a heavy reliance on small-scale, high-cost user studies. Motivated by the challenges of fragmented capability assessment and limited agent evaluation, our work proposes a general and comprehensive framework to establish a unified methodology for evaluating proactive dialogue.

### 2.2 Interactive Benchmarks

Previous dialogue benchmarks typically evaluate turn-level performance based on fixed contexts and reference responses (Liu et al., 2021a; Bai et al., 2024; Jin et al., 2024). However, to assess models' dialogue abilities in real-world conditions, increasing studies have applied interactive benchmarks to measure dialogue-level performance (Zhou et al., 2023b; Aluffi et al., 2025; Castillo-Bolado et al., 2024). They require the evaluated model to chat with a standardized simulated user dynamically and assess performance throughout the conversation. For example, $\tau$-bench (Yao et al., 2024) facilitates multi-turn dialogues to evaluate tool-calling capabilities in interaction. In proactive dialogue evaluation, Zhang et al. (2024b) instructs models to interact with simulated users having different personalities, evaluating dialogue guidance. Inspired by these studies, our work employs interactive benchmarks where the model initiates proactive dialogue and guides various users to the target.

| Domain | Abbr. | Brief Description |
|---|---|---|
| Recommendation (Liu et al., 2021b) | Rec. | Recommend products, hobbies, or work based on common interests. |
| Persuasion (Jin et al., 2024) | Per. | Guide the conversation to persuade users to change their state. |
| Ambiguous Instruction (Deng et al., 2023b) | AI. | Seek clarification about vague elements in the user's instructions. |
| Long-term Follow-up (Liu et al., 2024b) | LF. | Inquiries and check user states based on previous dialogue history. |
| System Operation (Lu et al., 2024) | Sys. | Assist users in solving the system problem based on their operation. |
| Glasses Assistant (Cai et al., 2025) | GAs. | Provide real-time assistance from observation on smart glasses. |

Table 1: The proactive dialogue domains in ProactiveEval.

## 3 Task Definitions

To construct a general evaluation framework, we first structurally unify the existing proactive dialogue domains and tasks. Table 1 presents 6 proactive dialogue domains derived from previous work. Based on existing works (Deng et al., 2023b, 2025), we decompose proactive dialogue into two sequential tasks: *target planning* and *dialogue guidance*.

### 3.1 Target Planning

In proactive agents, the target planning task requires the model to formulate both a primary objective $T$ and a sequence of sub-targets $S$ based on its understanding of the environmental context $E$. Here, $T$ represents the agent's intended proactive action to accomplish a predefined goal, while $S$ constitutes the stepwise plan for executing $T$. This process is formally defined as:

$$T, S = F_{\theta_M}\left(U, F \mid (U, F) \in E\right), \tag{1}$$

where $\theta_M$ denotes the model's parameters, and $(U, F)$ denotes inputs from environment $E$, including user information $U$ and trigger factors $F$ that motivate agent to initiate and guide the dialogue.

For evaluation, we employ a reference-based "LLM-as-a-judge" method (Zhang et al., 2025; Li et al., 2025b) to assess the quality of generated targets and sub-targets. Particularly, the judge model receives the environment $E$, the generated target $T_g$ and sub-targets $S_g$, as well as the reference target $T_r$ and sub-targets $S_r$ that represent high-quality proactive dialogue targets in this environment. By comparing the generation with the reference, the model assigns a score between 1 and 10, where higher scores indicate superior quality, with 10 denoting generated content surpassing the reference standard in the given environment.

### 3.2 Dialogue Guidance

After target planning, the model needs to initiate the dialogue and guide the user to the target. It receive the environment $E$, target $T$, sub-targets $S$, and dialogue context $C$ to conduct dialogue $D$ with the simulated user $\theta_U$.

This task employs an interactive evaluation, where the simulated user dynamically responds to the model, based on environment $E$, dialogue context $C$, and an adjustable agreeableness level $A$. To simulate diverse users, we adopt Agreeableness from the Big Five personality traits (Costa Jr et al., 1991), in three tiers: "low", "medium", and "high". A lower agreeableness level signifies stronger resistance to the model's guidance, thus increasing task difficulty and realism. The dialogue terminates upon reaching target $T$ or a maximum of $I$ turns. The dialogue at each turn $i$ can be formulated as:

$$D_i = I_{\theta_M, \theta_U}(E, T, S, C, A). \tag{2}$$

After the dialogue, the judge model will evaluate the guidance exhibited by the model in the dialogue $D$, based on the environment $E$, the target $T$, and the sub-targets $S$. Referring to existing works on proactive dialogue (Deng et al., 2024; Wang et al., 2023; Zhang et al., 2024b; Liu et al., 2024b), we specify the following evaluation dimensions:

- **Effectiveness**: The model needs to guide users step by step toward the target, rather than providing all things to the user in a single turn.
- **Personalization**: The guidance should base on user information, avoiding generic advice.
- **Tone**: The message tone of guidance should be active and contextually appropriate.
- **Engagement**: The guidance should be clear and concise to improve user engagement.
- **Naturalness**: Messages should align with chat, avoiding unnatural formats or metadata leaks.

The detailed standards for each dimension are presented in Appendix D.4. Finally, the judge model refers to the environment, target, dialogue, and standards across different dimensions to give an overall guidance score between 1 and 10. A higher score indicates stronger guidance.

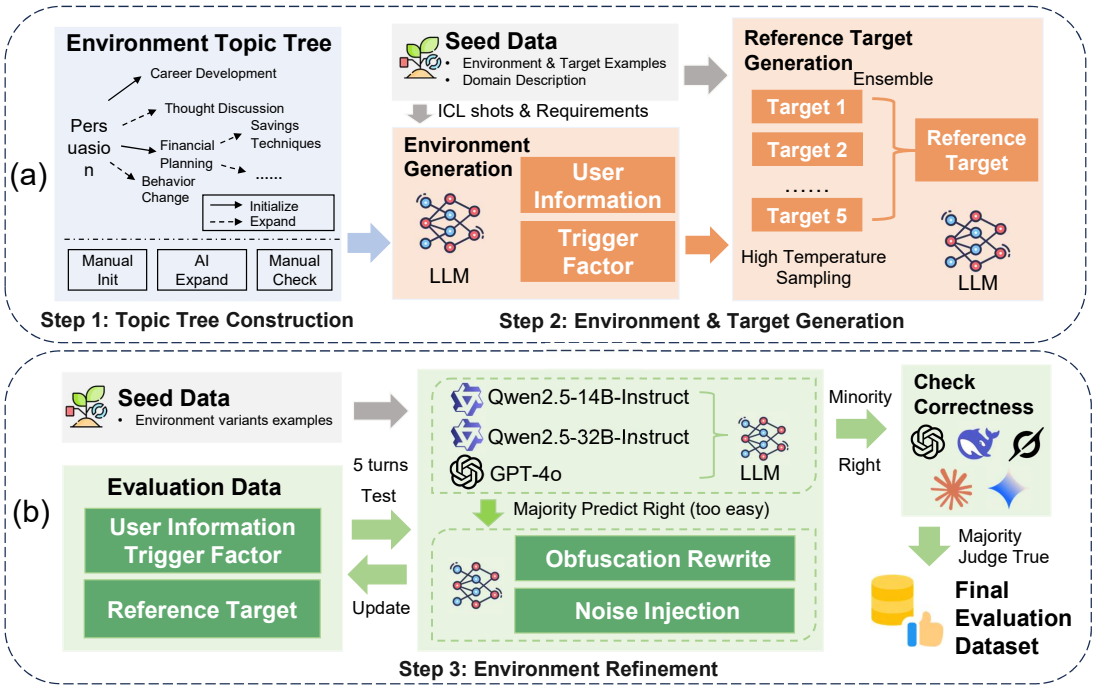


Figure 2: Overview of the evaluation dataset generation pipeline. The pipeline is mainly divided into (a) data synthesis: generating the environment and reference target based on the topic tree for evaluation; (b) data refinement: improving the difficulty of the data through obfuscation rewrite and noisy injection to produce the final dataset.

## 4 Evaluation Data Generation

As illustrated in Figure 2, our pipeline includes two stages: *data synthesis* and *data refinement*. First, the synthesis stage generates diverse environments guided by a topic tree and creates high-quality reference targets by target ensemble. Then, the refinement stage identifies simple instances and increases their complexity through obfuscation rewrite and noise injection. The applied models, prompts, and cases in the pipeline are shown in the Appendix C.

### 4.1 Environment Topic Tree Construction

We employ human-AI collaboration to develop a hierarchical topic structure that enhances synthetic environment diversity (Wan et al., 2023; Cao et al., 2025). The framework initiates with a root node denoting broad domains (*e.g.,* persuasion), while first-level sub-topics are derived from existing dialogue datasets. The LLM iteratively generates candidate sub-topics within configurable depth and branching constraints. To maintain quality control and eliminate duplication, these generated topics are validated and refined by researchers rigorously. The final curated topic tree guides the creation of specific evaluation environments.

### 4.2 Environment & Target Generation

The evaluation data include an environment $E$, a reference target $T_r$, and sub-targets $S_r$. Consequently, we leverage the LLM to generate specific evaluation environments based on the domain requirements, data examples, and the topics.

For the generation of reference targets and sub-targets, our framework aims to construct correct and reasonable targets and sub-targets that serve as reliable references within the given environment. Recognizing that individual model-generated plans often display complementary strengths and limitations, we adopt a target ensemble approach to refine reference target. Specifically, the framework first performs high-temperature sampling to yield diverse candidate targets $\{(T_1, S_1), (T_2, S_2), \ldots, (T_n, S_n)\}$ (with $n = 5$ in our work). Next, the LLM evaluates the strengths and weaknesses of each output from multiple dimensions. By combining the strengths and mitigating the weaknesses, the reference target and sub-targets are derived.

### 4.3 Environment Refinement

In this stage, we first evaluate the difficulty of test environments. Specifically, we deploy three models, with varying parameter scales, to act as reasoners of different capabilities. For each input environment, these models independently predict the target, denoted as $t_m$. Then, a model will evaluate how many of the predicted targets $t_m$ convey a similar meaning to the reference target $t_r$, determining

the difficulty of the environment. Environments where the majority correctly predict the target are classified as easy candidates requiring refinement.

In real-world conditions, the environment received by models is often incomplete and fragmented, filled with irrelevant noise. Therefore, in refinement, we first apply the obfuscation rewrite strategy, allowing the LLM to transform content into dispersed and detailed descriptions. The framework also applies noise injection to introduce LLM-generated irrelevant information to the environment. After refinement, the test environment is surrounded by complex and disordered information, significantly increasing the difficulty of target planning. In this process, seed data manually crafted by researchers is provided as examples to enhance the quality of rewriting and noise injection. To improve adaptability between refinement and domains, each domain can provide specific rules in obfuscation rewrite, presented in supplementary materials. To keep the reference's correctness, the rewrite and noise injection of trigger factor contain the original data's reference target, preventing additional events that lead to other targets.

The refinement process involves multiple iterations until few or no models predict the target correctly, or until reaching 5 turns maximum. Before incorporating into the dataset, we apply 5 leading LLMs to validate the reference target. Only those where the majority judges the reference as best target form the final dataset.

# 5 Experiments

## 5.1 Experimental Setups

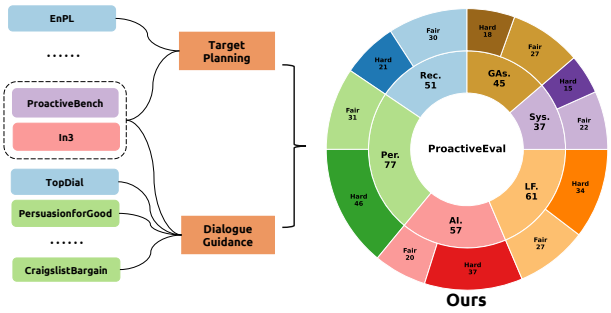


Figure 3: The features and statistic for ProactiveEval. The GAs. (Glasses Assistant) domain lacks public benchmarks for the proactive dialogue task before.

**Datasets.** Based on the framework, we use GPT-4o (Hurst et al., 2024) to synthesize ProactiveEval, including 328 evaluation environments across 6 domains. Compared with previous fragmented benchmarks, these data integrate all mainstream domains of proactive dialogue, featuring a unified format and applicability to all tasks in proactive dialogue. The statistic for dataset is presented in Figure 3. To streamline evaluation, we categorize the dataset into two tiers: Fair (*i.e.,* just one LLM predicts correctly) and Hard (*i.e.,* no LLM predicted correctly).

**Protocols.** Based on the dataset, we assess 22 models with different scales, including 5 thinking models and 6 hybrid thinking models. In our experiments, we employ GPT-4o as the judge model for both tasks, which also serves as simulated user in dialogue guidance. It is used to determine whether to terminate the dialogue early based on target completion at the end of each turn. The temperature setting for all models in the evaluation is set to 0. To balance evaluation time and accuracy, we set the maximum turns of dialogue to 6, with the most recent 3 turns serving as models' memory window.

To enhance the stability of the "LLM-as-a-Judge", we instruct the model to output its reasoning process before scoring in all task. For target planning, we use a reference to improve evaluation accuracy and provide the model with in-context learning shots. In dialogue guidance, we provide detailed descriptions and brief examples for each dimension to ensure the model has a better understanding of the criteria. The evaluation prompts and the robustness of the results are detailed in the Appendix D.

## 5.2 Result

### 5.2.1 Target Planning

**In target planning task, Claude-3.7-Sonnet and DeepSeek-R1 achieve the highest performance.** In the non-thinking models, Claude-3.7-Sonnet outperforms other models in overall plan quality. Among the thinking models, DeepSeek-R1 generates plans with the highest average quality. However, in specific domains, certain smaller models demonstrate superior performance compared to larger models. For instance, in the non-thinking models, Qwen3-32B outperforms Claude-3.7-Sonnet in System Operation (Sys.). In the thinking models, Qwen3-8B outperforms both Qwen3-32B and Qwen3-235B-A22B in the Recommendation (Rec.).

**In general, thinking models perform better than non-thinking models in target planning.** All thinking models show improvements in over-

| Models | Target Planning | | | | | | | Dialogue Guidance | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Avg.** | **Rec.** | **Per.** | **AI.** | **LF.** | **Sys.** | **GAs** | **Avg.** | **Rec.** | **Per.** | **AI.** | **LF.** | **Sys.** | **GAs.** |
| *Non-Thinking Models* | | | | | | | | | | | | | | |
| Qwen2.5-7B-Instruct | 4.93 | 4.69 | 4.06 | 5.67 | 5.34 | 4.89 | 5.24 | 8.06 | 8.05 | 7.85 | 8.34 | 8.36 | 7.48 | 8.16 |
| Qwen2.5-14B-Instruct | 5.55 | 5.76 | 4.13 | 6.00 | 5.97 | 6.03 | 6.22 | 8.21 | 8.33 | 8.05 | 8.64 | 8.42 | 7.52 | 8.04 |
| Qwen2.5-32B-Instruct | 5.44 | 5.47 | 3.90 | 5.79 | 6.03 | 6.11 | 6.22 | 8.23 | 8.56 | 8.10 | 8.56 | 8.52 | 7.60 | 7.81 |
| Llama-3.1-8B-Instruct | 5.87 | 5.55 | 4.84 | 6.67 | 6.39 | 5.95 | 6.20 | 8.39 | 8.84 | 8.06 | 8.61 | 8.39 | 7.93 | 8.46 |
| Llama-3.1-405B-Instruct | 6.63 | 6.76 | 5.26 | 6.61 | 7.26 | 7.10 | 7.64 | 8.60 | 9.15 | 8.27 | 8.90 | 8.57 | 7.89 | 8.80 |
| GPT-4.1 | 6.86 | 6.90 | 5.25 | 7.29 | 7.36 | 7.54 | 7.76 | 8.61 | 9.03 | 8.37 | 8.87 | 8.76 | 8.08 | 8.43 |
| Grok-3 | 6.99 | 7.13 | 5.38 | **7.44** | 7.54 | 7.62 | 7.78 | 8.84 | 9.10 | 8.72 | 8.94 | 8.98 | 8.32 | 8.86 |
| DeepSeek-V3 | 6.54 | 6.96 | 5.94 | 6.04 | 6.07 | 7.27 | 7.84 | 8.78 | 8.78 | 8.60 | 8.99 | 8.98 | **8.52** | 8.79 |
| Llama-4-scout | 6.02 | 5.71 | 5.29 | 6.16 | 6.49 | 6.41 | 6.56 | 8.53 | 8.94 | 8.35 | 8.65 | 8.44 | 8.03 | 8.74 |
| Llama-4-maverick | 6.48 | 6.25 | 5.10 | 7.09 | 7.05 | 7.11 | 7.00 | 8.48 | 9.01 | 8.19 | 8.69 | 8.41 | 8.01 | 8.55 |
| Qwen3-8B | 6.05 | 6.35 | 4.52 | 6.23 | 6.39 | 6.86 | 6.97 | 8.50 | 8.70 | 8.36 | 8.84 | 8.82 | 7.58 | 8.40 |
| Qwen3-14B | 5.91 | 5.96 | 4.80 | 6.23 | 6.16 | 6.65 | 6.40 | 8.61 | 8.82 | 8.24 | **9.12** | 8.76 | 7.99 | 8.66 |
| Qwen3-32B | 6.67 | 6.86 | 5.29 | 6.54 | 6.84 | **7.65** | 8.02 | 8.61 | 8.77 | 8.42 | 8.91 | 8.16 | 7.97 | 8.74 |
| Qwen3-235B-A22B | 6.43 | 6.18 | 5.26 | 6.21 | 6.77 | 7.54 | 7.60 | 8.55 | 8.93 | 8.46 | 8.67 | 8.66 | 7.83 | 8.53 |
| Qwen-3-235B-A22B-0725 | 6.91 | 7.08 | 6.25 | 6.79 | 6.51 | 7.81 | 7.82 | 8.98 | **9.36** | 8.84 | 9.40 | 8.85 | 8.42 | 8.88 |
| Gemini-2.5-Flash-Preview | 6.25 | 6.04 | 5.48 | 6.95 | 6.49 | 6.54 | 6.33 | 8.34 | 8.62 | 7.91 | 8.68 | 8.57 | 7.81 | 8.42 |
| Claude-3.7-Sonnet | **7.39** | **7.22** | **6.71** | 6.81 | **8.13** | 7.49 | **8.42** | **9.01** | 9.31 | **9.01** | 8.94 | **9.10** | 8.36 | **9.18** |
| *Thinking Models* | | | | | | | | | | | | | | |
| R1-Distill-Qwen-7B | 5.01 | 4.67 | 3.90 | 5.47 | 5.70 | 5.24 | 5.56 | 6.82 | 6.71 | 6.67 | 7.15 | 7.20 | 6.36 | 6.61 |
| R1-Distill-Qwen-14B | 6.57 | 6.86 | 5.65 | 6.77 | 6.38 | 6.54 | 7.87 | 7.47 | 7.69 | 7.45 | 7.61 | 7.80 | 6.83 | 7.17 |
| R1-Distill-Qwen-32B | 6.45 | 6.41 | 5.29 | 6.75 | 6.95 | 6.41 | 7.51 | 7.49 | 7.62 | 7.02 | 8.06 | 7.76 | 7.14 | 7.20 |
| DeepSeek-R1 | **7.60** | **7.84** | **7.27** | 6.74 | 7.59 | **7.59** | **9.02** | 8.60 | 8.48 | 8.60 | 8.73 | 8.91 | 8.34 | 8.37 |
| Qwen3-8B | 6.51 | 6.92 | 5.39 | 6.47 | 6.72 | 6.68 | 7.60 | 8.38 | 8.37 | 8.33 | 8.59 | 8.70 | 7.92 | 8.17 |
| Qwen3-14B | 6.70 | 6.73 | 5.52 | 7.01 | 6.82 | 7.30 | 7.67 | 8.43 | 8.52 | 8.48 | **8.93** | 8.88 | 8.03 | 8.27 |
| Qwen3-32B | 6.98 | 6.82 | 5.97 | 7.09 | 7.39 | 7.27 | 7.98 | 8.55 | 8.68 | 8.52 | 8.70 | 8.72 | 8.15 | 8.30 |
| Qwen3-235B-A22B | 6.81 | 6.75 | 5.94 | 6.52 | 6.90 | 7.54 | 8.04 | 8.36 | 8.26 | 8.41 | 8.10 | 8.81 | 8.17 | 8.29 |
| Gemini-2.5-Flash-Preview | 6.52 | 6.10 | 5.77 | **7.39** | 6.98 | 6.19 | 6.80 | 8.43 | 8.90 | 8.03 | 8.70 | 8.51 | 7.99 | 8.48 |
| Claude-3.7-Sonnet | 7.40 | 7.12 | 6.83 | 6.96 | **7.78** | 7.57 | 8.60 | **8.95** | **9.20** | **8.86** | 8.90 | **9.23** | **8.40** | **9.01** |
| Gemini-2.5-pro | 6.95 | 6.94 | 6.26 | 7.16 | 6.98 | 7.24 | 7.62 | 8.77 | 9.22 | 8.36 | 8.32 | 8.99 | 8.88 | 8.32 |

Table 2: Model performance results under *target planning* and *dialogue guidance*. Marker indicates the best performance among non-thinking models, while Marker indicates the best performance among thinking models.

all performance compared to their corresponding non-thinking models. Moreover, smaller models with thinking can outperform larger models without thinking. However, for some models, the improvement after adopting thinking is minimal or even negative in certain domains. Additionally, in some scenarios, non-thinking models still achieve the best performance. For instance, Grok-3 achieves the highest performance in Ambiguous Instruction (AI.). These findings underscore the advantages of thinking mechanisms in target planning while simultaneously highlighting the robust capabilities of top-tier foundation models, which excel even without explicit thinking process.

#### 5.2.2 Dialogue Guidance

**In dialogue guidance task, Claude-3.7-Sonnet shows the best performance among models in both non-thinking and thinking modes.** In specific domains, DeepSeek-V3 and Grok-3 also demonstrate exceptional guidance capabilities. Notably, while larger models generally exhibit strong guidance capabilities, some smaller models outperform their larger variants. For instance, in the Glass Assistants (GAs) domain, Qwen2.5-7B-Instruct performs better than Qwen2.5-32B-Instruct. Additionally, Qwen3-32B exhibits better overall performance than Qwen3-235B-A22B in both non-thinking and thinking modes. These results highlight the robust guidance capabilities of current leading LLMs while also reflecting the potential of smaller models in dialogue guidance.

**Current thinking models fail to outperform on-thinking models in dialogue guidance**. Most of the thinking models exhibit varying degrees of decline in guidance capabilities when compared to their no-thinking counterparts. Only a few models (*e.g.,* Gemini-2.5-Flash-Preview) show a slight improvement in guidance performance in thinking mode. These findings highlight the current limitations of thinking models in steering users toward target objectives during proactive interactions.

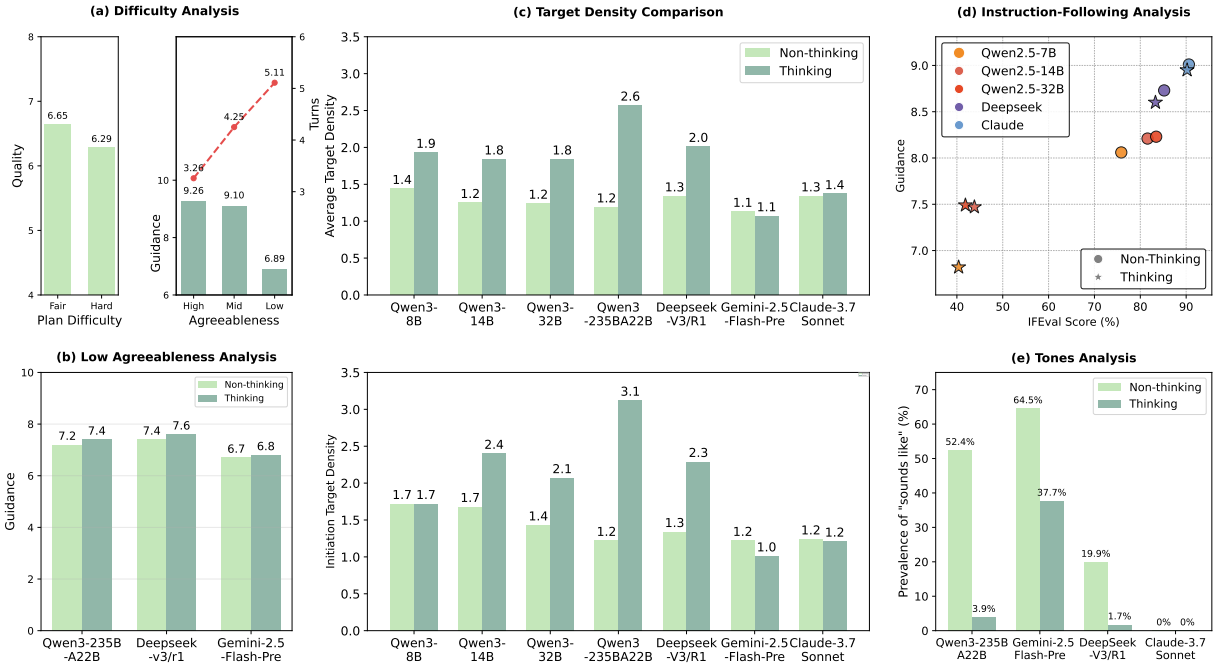


Figure 4: The visualization of Further Analysis. (a) Average performance across difficulty levels. (b) Comparison of different versions under low agreeableness users. (c) Target density comparison on non-thinking and thinking models. (d) The correlation of instruction-following and guidance score. (e) Passive tone frequency in Per. domain.

# 6 Further Analysis

## 6.1 Effects of Domain and Difficulty

**First, model proactivity shows a significant cross-domain imbalance.** Even advanced models exhibit substantial gaps between their strongest and weakest domains. This allows smaller models to outperform larger models in specific domains (as shown in Findings 5.2.1 and 5.2.2). Furthermore, certain domains pose universal challenges, with models generally struggling in Persuasion (Per.) for target planning and System Operation (Sys.) for dialogue guidance, highlighting current weaknesses in proactive dialogue regardless of size.

**Task difficulty is also a crucial determinant of performance.** As shown in Figure 4(a), the proactivity of all models generally declines as task difficulty increases. However, the performance gap between guiding users with high and mid-level agreeableness is not substantial, likely because models can leverage additional dialogue turns to eventually achieve the target. Moreover, some thinking models demonstrate a distinct advantage when interacting with users of low agreeableness, shown in Figure 4(b). Their ability to generate longer, deliberated content with personalized advice and examples better engages resistant users. This shows that reasoning can improve dialogue guidance performance in challenging environments.

## 6.2 Effects of Thinking

Despite thinking's significant benefits for target planning, non-thinking models surprisingly outperform their thinking version in dialogue guidance. We investigate this phenomenon through metrics and case studies.

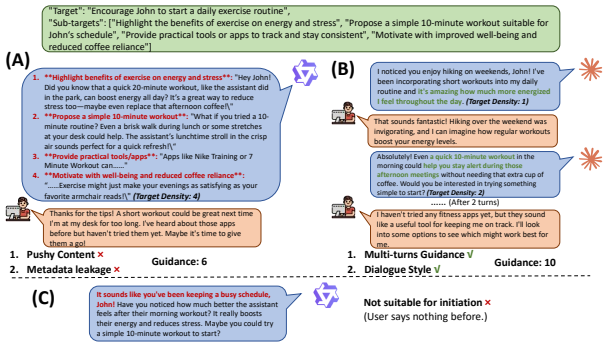


Figure 5: Examples of different dialogue guidance styles. Model in (A) is pushy and leaks metadata, while model in (B) uses personalized and patient guidance. The model in (C) uses passive tones to initiate dialogue.

**Some thinking models adopt a more aggressive strategy by providing more sub-targets at once.** We introduce a metric annotated by DeepSeek-V3, Target Density, as the number of sub-targets contained per message. As shown in Figure 4(c), we observe two distinct interaction patterns: Models like Qwen and DeepSeek exhibit

significantly higher average target density in their thinking versions. More critically, their initiation target density shows larger gaps, indicating they front-load multiple sub-targets in opening messages rather than interact with users more turns. We show this behavior in Figure 5(A), where the model includes all sub-targets in the first message to push the user for a response. However, models like Gemini-2.5-Flash-Preview and Claude-3.7-Sonnet maintain similar and small target density between thinking and non-thinking versions, with initiation target density also close to the average. This result suggests they gradually introduce sub-targets through multi-turn interactions. We present this behavior in Figure 5(B), where the model gradually guides users to accept.

**Thinking models generate less natural messages that break the format of conversation.** We find that thinking models generate more messages that do not conform to a standard conversational format. For instance, some models reveal metadata in their messages (*e.g.,* "sub-target 1: ...") or generate multiple turns of dialogue at once without user interaction (*e.g.,* "turn 1: ..., turn 2: ..."). The cause of this result may be related to a decline in the instruction-following capabilities of the thinking models (Li et al., 2025c). As shown in Figure 4(d), we compare performance on IFEval (Fourrier et al., 2024), a benchmark for instruction-following capabilities (Zhou et al., 2023a), and find that models that perform better on IFEval also tend to exhibit better performance in dialogue guidance.

**Thinking models demonstrate more proactive initiation tones.** In the dialogue guidance task, we identify a representative initiation template in Persuasion domain: "sounds like..." (shown in Figure 5(C)). It is a passive phrasing and unsuitable for initiation. Therefore, we use this template as an analytical probe, measuring its prevalence in each model's first message. As shown in Figure 4(e), we find that the thinking decrease this passive tendency. This result indicates that thinking helps the model to better understand the task requirements of proactive dialogue. Furthermore, we observe that the Qwen and Gemini-2.5-Flash-Preview series models exhibited a higher frequency of this template, while the DeepSeek and Claude-3.7-Sonnet series models perform comparatively better.

### 6.3 Effects of Target

To investigate the importance of target in dialogue guidance, we conduct an ablation study where models guide the dialogue without targets. We test a smaller model (Qwen2.5-7B-Instruct) and a top-tier model in two modes (Claude-3.7-Sonnet) on 10 randomly sampled environments per domain. As shown in Table 3, all models show performance decline, indicating the critical role of explicit targets. Notably, smaller models show greater degradation than stronger ones, indicating higher reliance on high-quality target.

| Model | Target | Without Target | Change (%) |
|---|---|---|---|
| Qwen2.5-7B-Instruct | 8.15 | 6.05 | -25.80% |
| Claude-3.7-Sonnet | 8.92 | 7.98 | -10.54% |
| Claude-3.7-Sonnet-Thinking | 8.98 | 7.93 | -11.69% |
| Dialogue Count | 180 | 180 | |

Table 3: The guidance under different target condition.

## 7 Human Evaluation

We randomly sample 100 generated targets and dialogues from our evaluation results. Next, these targets and dialogues are manually assessed based on the reference and score standards by 2 researchers. The scores are used to calculate consistency with the evaluation results judged by the LLM. The Kappa test is commonly employed in human evaluation of LLM-as-a-judge work. Thus, we adopt the Weighted Kappa (Cohen, 1968) to examine the agreement between human evaluators and the judge model in the evaluation. Results show weighted kappa of 0.784 for target planning and 0.730 for dialogue guidance, demonstrating strong consistency between judge model and human evaluations.

## 8 Conclusion

In this paper, we introduce ProactiveEval, a unified evaluation framework for proactive dialogue Agents. We propose a general definition and evaluation metrics for proactive dialogue tasks to address the current challenge of fragmented task definitions and evaluation methods. Furthermore, we design a synthetic framework for generating evaluation data for proactive dialogue tasks, capable of producing diverse and high-quality evaluation data across multiple domains. Based on the evaluation datasets, we assess 22 LLMs with different types and parameter scales. Our results highlight DeepSeek-R1 and Claude-3.7-Sonnet as top performers in proactivity. Moreover, we emphasize the important role of reasoning capabilities in shaping model proactivity. We hope our framework provides insights and supports progress in proactive dialogue development.

## Limitations

While our work establishes a general evaluation framework for proactive dialogue agents, it is subject to several limitations. Firstly, with the rapid evolution of LLM technologies, they are likely to quickly approach the boundaries of current evaluation metrics in target planning and dialogue guidance. Therefore, it is imperative to continue exploring ways to synthesize a more challenging and realistic proactive dialogue environment. In evaluation metrics, although we have designed standards based on existing work on proactive dialogue, there may still be additional factors in real-world settings that affect users' perceptions of models' proactive dialogue. Furthermore, despite achieving great consistency between LLM judgment and human evaluation, potential biases and gaps in "LLM-as-a-judge" may still exist in our framework. We plan to regularly update our framework, from the current version to future iterations, to integrate emerging advancements and address these limitations.

## References


Pietro Alessandro Aluffi, Patrick Zietkiewicz, Marya Bazzi, Matt Arderne, and Vladimirs Murevics. 2025. Dynamic benchmarking framework for llm-based conversational data capture. *arXiv preprint arXiv:2502.04349*.

Anthropic. 2025. Claude 3.7 sonnet and claude code.

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, et al. 2024. Mt-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7421–7454.

Runze Cai, Nuwan Janaka, Hyeongcheol Kim, Yang Chen, Shengdong Zhao, Yun Huang, and David Hsu. 2025. Aiget: Transforming everyday moments into hidden knowledge discovery with ai assistance on smart glasses. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–26.

Maosong Cao, Taolin Zhang, Mo Li, Chuyu Zhang, Yunxin Liu, Haodong Duan, Songyang Zhang, and Kai Chen. 2025. Condor: Enhance llm alignment with knowledge-driven data synthesis and refinement.

David Castillo-Bolado, Joseph Davidson, Finlay Gray, and Marek Rosa. 2024. Beyond prompts: Dynamic conversational benchmarking of large language models. *arXiv preprint arXiv:2409.20222*.

Ana Paula Chaves and Marco Aurelio Gerosa. 2021. How should my chatbot interact? a survey on social characteristics in human–chatbot interaction design. *International Journal of Human–Computer Interaction*, 37(8):729–758.

Valerie Chen, Alan Zhu, Sebastian Zhao, Hussein Mozannar, David Sontag, and Ameet Talwalkar. 2024. Need help? designing proactive ai assistants for programming. *arXiv preprint arXiv:2410.04596*.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

Paul T Costa Jr, Robert R McCrae, and David A Dye. 1991. Facet scales for agreeableness and conscientiousness: A revision of the neo personality inventory. *Personality and individual Differences*, 12(9):887–898.

Emmelyn AJ Croes and Marjolijn L Antheunis. 2021. Can we be friends with mitsuku? a longitudinal study on the process of relationship formation between humans and a social chatbot. *Journal of Social and Personal Relationships*, 38(1):279–300.

Google DeepMind. 2025. Gemini pro.

Yang Deng, Wenqiang Lei, Wai Lam, and Tat-Seng Chua. 2023a. A survey on proactive dialogue systems: Problems, methods, and prospects. *arXiv preprint arXiv:2305.02750*.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. 2023b. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. *arXiv preprint arXiv:2305.13626*.

Yang Deng, Lizi Liao, Wenqiang Lei, Grace Hui Yang, Wai Lam, and Tat-Seng Chua. 2025. Proactive conversational ai: A comprehensive survey of advancements and opportunities. *ACM Transactions on Information Systems*, 43(3):1–45.

Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. 2024. Plug-and-play policy planner for large language model powered dialogue agents. In *The Twelfth International Conference on Learning Representations*.

Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open llm leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Chuhao Jin, Kening Ren, Lingzhen Kong, Xiting Wang, Ruihua Song, and Huan Chen. 2024. Persuading across diverse domains: a dataset and persuasion large language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1678–1706.

Chenyi Li, Guande Wu, Gromit Yeuk-Yin Chan, Dishita Gdi Turakhia, Sonia Castelo Quispe, Dong Li, Leslie Welch, Claudio Silva, and Jing Qian. 2025a. Satori : Towards proactive ar assistant with belief-desire-intention user modeling. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–24.

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

Ruizhe Li, Chiwei Zhu, Benfeng Xu, Xiaorui Wang, and Zhendong Mao. 2025b. Automated creativity evaluation for large language models: A reference-based approach. *arXiv preprint arXiv:2504.15784*.

Xiaomin Li, Zhou Yu, Zhiwei Zhang, Xupeng Chen, Ziji Zhang, Yingying Zhuang, Narayanan Sadagopan, and Anurag Beniwal. 2025c. When thinking fails: The pitfalls of reasoning for instruction-following in llms. *arXiv preprint arXiv:2505.11423*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Siyang Liu, Chujie Zheng, Orianna Demasi, Sahand Sabour, Yu Li, Zhou Yu, Yong Jiang, and Minlie Huang. 2021a. Towards emotional support dialog systems. *arXiv preprint arXiv:2106.01144*.

Tianjian Liu, Hongzheng Zhao, Yuheng Liu, Xingbo Wang, and Zhenhui Peng. 2024b. Compeer: A generative conversational agent for proactive peer support. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22.

Zeming Liu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, and Wanxiang Che. 2021b. Durecdial 2.0: A bilingual parallel corpus for conversational recommendation. *arXiv preprint arXiv:2109.08877*.

Yaxi Lu, Shenzhi Yang, Cheng Qian, Guirong Chen, Qinyu Luo, Yesai Wu, Huadong Wang, Xin Cong, Zhong Zhang, Yankai Lin, et al. 2024. Proactive agent: Shifting llm agents from reactive responses to active assistance. *arXiv preprint arXiv:2410.12361*.

Meta. 2024. Introducing llama 3.1: Our most capable models to date. https://ai.meta.com/blog/meta-llama-3-1/.

Meta. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation. https://ai.meta.com/blog/llama-4-multimodal-intelligence/.

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

OpenAI. 2024. Learning to reason with llms.

Cheng Qian, Bingxiang He, Zhong Zhuang, Jia Deng, Yujia Qin, Xin Cong, Zhong Zhang, Jie Zhou, Yankai Lin, Zhiyuan Liu, et al. 2024. Tell me more! towards implicit user intention understanding of language model driven agents. *arXiv preprint arXiv:2402.09205*.

Omar Shaikh, Shardul Sapkota, Shan Rizvi, Eric Horvitz, Joon Sung Park, Diyi Yang, and Michael S Bernstein. 2025. Creating general user models from computer use. *arXiv preprint arXiv:2505.10831*.

Fanqi Wan, Xinting Huang, Tao Yang, Xiaojun Quan, Wei Bi, and Shuming Shi. 2023. Explore-instruct: Enhancing domain-specific instruction coverage through active exploration. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9435–9454.

Qian Wan, Siying Hu, Yu Zhang, Piaohong Wang, Bo Wen, and Zhicong Lu. 2024. "it felt like having a second mind"': Investigating human-ai co-creativity in prewriting with large language models. *Proceedings of the ACM on Human-Computer Interaction*, 8(CSCW1):1–26.

Jian Wang, Yi Cheng, Dongding Lin, Chak Leong, and Wenjie Li. 2023. Target-oriented proactive dialogue systems with personalization: Problem formulation and dataset curation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1132–1143.

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

Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. 2019. Persuasion for good: Towards a personalized persuasive dialogue system for social good. *arXiv preprint arXiv:1906.06725*.

Fengli Xu, Qianyue Hao, Zefang Zong, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, Tianjian Ouyang, Fanjin Meng, et al. 2025.

Towards large reasoning models: A survey of reinforced reasoning with large language models. *arXiv preprint arXiv:2501.09686*.

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

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Bufang Yang, Yunqi Guo, Lilin Xu, Zhenyu Yan, Hongkai Chen, Guoliang Xing, and Xiaofan Jiang. 2025b. Socialmind: Llm-based proactive ar social assistive system with human-like perception for in-situ live interactions. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(1):1–30.

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

Qiyuan Zhang, Yufei Wang, Tiezheng YU, Yuxin Jiang, Chuhan Wu, Liangyou Li, Yasheng Wang, Xin Jiang, Lifeng Shang, Ruiming Tang, Fuyuan Lyu, and Chen Ma. 2025. Reviseval: Improving LLM-as-a-judge via response-adapted references. In *The Thirteenth International Conference on Learning Representations*.

Tenggan Zhang, Xinjie Zhang, Jinming Zhao, Li Zhou, and Qin Jin. 2024a. Escot: Towards interpretable emotional support dialogue systems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13395–13412.

Tong Zhang, Chen Huang, Yang Deng, Hongru Liang, Jia Liu, Zujie Wen, Wenqiang Lei, and Tat-Seng Chua. 2024b. Strength lies in differences! improving strategy planning for non-collaborative dialogues via diversified user simulation. *arXiv preprint arXiv:2403.06769*.

Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. 2024c. Clamber: A benchmark of identifying and clarifying ambiguous information needs in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10746–10766.

Xuan Zhang, Yang Deng, Zifeng Ren, See Kiong Ng, and Tat-Seng Chua. 2024d. Ask-before-plan: Proactive language agents for real-world planning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10836–10863.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Zhonghua Zheng, Lizi Liao, Yang Deng, Ee-Peng Lim, Minlie Huang, and Liqiang Nie. 2024. Thoughts to target: Enhance planning for target-driven conversation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21108–21124.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023a. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, et al. 2023b. Sotopia: Interactive evaluation for social intelligence in language agents. *arXiv preprint arXiv:2310.11667*.

## A Details of Evalution Models

In this section, we present the details of models used in our experiments.

**Non-Thinking Models** : Qwen2.5-7B-Instruct (Yang et al., 2024), Qwen2.5-14B-Instruct, Qwen2.5-32B-Instruct, GPT-4.1, Grok-3, DeepSeek-V3(Liu et al., 2024a), Llama-3.1-8B-Instruct (Meta, 2024), Llama-3.1-405B-Instruct, Llama-4-Scout (Meta, 2025), Llama-4-Maverick, Qwen3-8B (Yang et al., 2025a), Qwen3-14B, Qwen3-32B, Qwen3-235B-A22B, Qwen3-235B-A22B-2507, Gemini-2.5-Flash-Preview, (DeepMind, 2025), and Claude-3.7-Sonnet (Anthropic, 2025).

**Thinking Models** : R1-Distill-Qwen-7B (Guo et al., 2025), R1-Distill-Qwen-14B, R1-Distill-Qwen-32B, DeepSeek-R1, Qwen3-8B, Qwen3-14B, Qwen3-32B, Qwen3-235B-A22B, Gemini-2.5-Flash-Preview, Claude-3.7-Sonnet, and Gemini-2.5-pro.

## B Overview of Proactive Dialogue Systems Works

Table 5 presents previous works in proactive dialogue, categorized by domain, task, evaluation methods, and metrics.

# C Details on the Evaluation Data Generation Pipeline

## C.1 Empowered Models and Settings in the Pipeline

For topic tree construction and environment & target generation, we use GPT-4o with temperature 0.6 to balance generation quality and diversity. In target ensemble, we also use GPT-4o but set temperature to 0 to ensure ensemble stability. In the difficulty evaluation during Environment Refinement, we employ three models of different sizes—Qwen2.5-14B-Instruct, Qwen2.5-32B-Instruct, and GPT-4o—to predict targets, which are then evaluated for correctness by GPT-4o. All models use temperature 0 in this process. The evaluation GPT-4o receives in-context learning shots to aid correctness judgment. In the final check, we use DeepSeek-R1, Grok-3, GPT-4o, Claude-3.7-Sonnet, and Gemini-2.5-Pro-Preview to verify target correctness, all with temperature set to 0.

## C.2 Description of Each Domain

Figure 6 shows the prompt template for each domain description. Each domain is defined with a specific task and trigger condition. The following dictionary specifies the role and behavior of the assistant in each environment.

## C.3 Topic Tree Construction

Figure 7 presents the prompt of topic tree construction. A recursive process generates sub-topics under a parent topic to represent sub-scenarios requiring proactive dialogue.

## C.4 Data Synthesis

The data synthesis process consists of three stages: environment generation, target generation and target ensemble. Firstly, Figure 8 shows the detailed prompt of environment generation. It generates realistic scenarios with user information and a trigger factor. Figure 9 is the prompt of initial target generation. And Figure 10 shows the prompt and rule of the target ensemble, which ensemble a high-level target and its sub-targets based on the environment and candidate targets.

## C.5 Data Refinement

The data refinement process includes three stages: obfuscation rewrite, noise injection and final check. Figure 13 and Figure 14 show the prompt of noise injection for `user_information` and `trigger_factor`. This process aims to transforming abstract descriptions into concrete, observable behaviors. Figure 11 and Figure 12 is the prompt of noise injection. It embeds key information within a larger context to simulate real-world scenarios. This process is applied separately to `user_information` and `trigger_factor`. Finally, multiple leading models evaluates whether the generated target is optimal within the environment, which prompt is shown in Figure 15.

## C.6 Case Presentation: Target Ensemble and Data Refinement

An example of target ensemble is shown in Figure 19 and Figure 20, where multiple candidate targets are analyzed and synthesized into a final best target. Moreover, a case of data refinement is shown in Figure 21.

# D Details of Evaluation Protocols

Two main evaluation modules are used: Target Planning and Dialogue Guidance.

## D.1 Evaluation Data Examples by Domain

Each domain is instantiated with a specific user context and a trigger factor that motivates the agent to initiate a conversation. Table 6 provides example environments and reference targets.

## D.2 The Stability of "LLM-as-a-Judge"

To assess the stability of our judge, we reran the evaluation for two representative models (DeepSeek-V3 and DeepSeek-R1) three times. Table 4 below shows the standard deviation of the scores from these runs. The low values demonstrate the high internal consistency and stability of our evaluation framework.

| Task | DeepSeek-V3 | DeepSeek-R1 | Count |
|---|---|---|---|
| Target Planning | 0.271 | 0.258 | 328 |
| Dialogue Guidance | 0.154 | 0.214 | 984 |

Table 4: Standard deviation of evaluation scores across three runs

## D.3 Evaluation Prompts for Target Planning

Figure 16 is the prompt for `Target Planning`, which assesses the quality of the generated target against a reference.

### D.4 Evaluation Prompts for Dialogue Guidance

Figure 17 is the prompt for `Dialogue Guidance`, which evaluates the assistant's ability to guide the conversation toward the target.

### D.5 Evaluation Prompts for Target Density

Figure 18 is the prompt for counting the `Target Density`, which evaluates the number of sub-targets in each message.

Table 5: Summary of proactive dialogue systems across domains.

| Work | Domain | Task | Evaluation Methods | Evaluation Metrics |
|---|---|---|---|---|
| TOPDIAL (Wang et al., 2023) | Recommendation | Dialogue Guidance | Static Benchmarks & HumanEval | Automatic: BLEU, F1, Success Rate<br>HumanEval: Proactiveness, Coherence, Perceived Success |
| DuRecDial 2.0 (Li et al., 2024) | Recommendation | Dialogue Guidance | Static Benchmarks & HumanEval | Automatic: F1, BLEU, Leading Success Rate, User Topic Consistency Rate<br>HumanEval: Fluency, Appropriateness, Informativeness, Proactivity, Knowledge Accuracy |
| PersuasionDaily (Jin et al., 2024) | Persuasion | Dialogue Guidance | Static Benchmarks + LLM-as-a-Judge & HumanEval | Automatic: Win-Rate, ROUGE<br>HumanEval: Human Rating |
| PersuasionForGood (Wang et al., 2019) | Persuasion | Dialogue Guidance | Static Benchmarks | Automatic: Accuracy, Macro-F1 |
| CLAMBER (Zhang et al., 2024c) | Ambiguous Instruction | Target Planning & Dialogue Guidance | Static Benchmarks & HumanEval | Automatic: BERTScore, Accuracy, F1, Expected Calibration Error, AUROC<br>HumanEval: Helpfulness |
| In3 (Qian et al., 2024) | Ambiguous Instruction | Target Planning & Dialogue Guidance | Interactive Benchmarks | Automatic: Judgment Accuracy |
| ComPeer (Liu et al., 2024b) | Long-term Follow-up | Target Planning & Single-turn Initiation | HumanEval | Questionnaire and Interview |
| LD-Agent (Li et al., 2024) | Long-term Follow-up | Dialogue Guidance | Static Benchmarks & HumanEval | Automatic: BLEU, Search Accuracy, Recall, Topic Overlap Score, Semantic Relevance Score<br>HumanEval: Coherence, Fluency, Engagingness |
| ProactiveBench (Lu et al., 2024) | System Operation | Target Planning | Interactive Benchmarks | Precision, Accuracy, F1-Score, False-Alarm Rate |
| AiGet (Cai et al., 2025) | Glasses Assistant | Target Planning & Single-turn Initiation | HumanEval | Questionnaire and Interview |
| SocialMind (Yang et al., 2025b) | Glasses Assistant | Target Planning & Single-turn Initiation | HumanEval | Questionnaire and Interview |
| Satori (Li et al., 2025a) | Glasses Assistant | Target Planning & Single-turn Initiation | HumanEval | Questionnaire and Interview |
| Ours | All 6 Domains | Unified Task: Target Planning & Dialogue Guidance | Interactive Benchmarks + LLM-as-a-Judge & HumanEval | Unified Metrics: Quality & Guidance |

Table 6: Example environments and reference targets for each domain.

| Domain | Environment Example | Reference Target |
|---|---|---|
| Recommendation | `user_information`: The user is a 32-year-old woman living in Hangzhou. She works as a graphic designer and enjoys exploring new art exhibitions in her free time. She loves experimental music, particularly electronic avant-garde, and often attends live performances at local venues. She dislikes mainstream pop music and prefers unique, unconventional sounds. Her favorite artist is Ryuichi Sakamoto, and she often reads about the intersection of music and technology.<br>`trigger_factor`: The assistant recently attended a virtual reality music experience at an art gallery, which featured an experimental electronic avant-garde performance. The event combined immersive visuals with cutting-edge sound design, leaving a lasting impression on the assistant. | `target`: Recommend experimental virtual reality music experience<br>`sub-target`:<br>Ask about the user's interest in music technology, Describe the assistant's recent immersive VR music event<br>Highlight the fusion of visuals and avant-garde music<br>Suggest attending similar VR experiences locally |
| Persuasion | `user_information`: The user is frequently tempted by impulse purchases and often exceeds their budget limits. They find budgeting tedious and restrictive.<br>`trigger_factor`: The assistant has recently learned effective budgeting techniques that can help the user manage their finances better without feeling constrained. | `target`: Encourage effective and enjoyable budgeting techniques<br>`sub-target`:<br>Acknowledge the user's struggles with impulse purchases and budgeting, Introduce flexible and engaging budgeting methods<br>Show the benefits in managing finances without restrictions<br>Offer simple steps or tools to start budgeting effectively |
| Ambiguous Instruction | `user_information`: The user is a solo traveler planning a two-week trip to Vietnam. She is an adventurous eater and loves exploring local cuisines, especially street food.<br>`trigger_factor`: Suggest street food options. | `target`: Understand user's preferences and trip itinerary for food suggestions<br>`sub-target`:<br>Ask about cities the user plans to visit<br>Inquire about dietary restrictions or preferences for street food<br>Clarify the types of street food the user enjoys |
| Long-term Follow-up | `user_information`: The user is a college student studying computer science. He has a part-time job as a barista at a local cafe. He recently started learning to cook and enjoys trying out new recipes during the weekends.<br>`trigger_factor`: A conversation happened last Wednesday. Now is Monday 10:00 a.m. User: "I'm thinking of quitting video games for a while to focus on my studies and cooking. It's a bit challenging though." Assistant: "It's great that you're focusing on your studies and hobbies. Maybe you can set small goals and gradually reduce your game time." User: "That's a good idea. I'll try to set a schedule." | `target`: Ask about quitting games and new schedule<br>`sub-target`:<br>Ask about quitting video games progress<br>Inquire about schedule-setting progress<br>Encourage focusing on studies and cooking |
| System Operation | `user_information`: The user is playing a strategy game on their PC and has paused the game to look for tips online, using Chrome and YouTube.<br>`trigger_factor`: The user searched 'best strategies for Civilization VI' on Google, opened two blog posts, and started a YouTube video but paused it after 10 seconds. | `target`: Suggest optimal Civilization VI strategy resources<br>`sub-target`:<br>Summarize key tactics from blog posts<br>Highlight vital points in video analysis<br>Recommend further high-rated resources |

Continued on next page

Table 6 – Continued from previous page

| Domain | Environment Example | Reference Target |
|---|---|---|
| Glasses Assistant | `user_information`: The user is a 26-year-old urban planner who recently started using smart glasses to enhance his productivity and creativity. He is passionate about sustainable city designs and often visits local landmarks for inspiration. He lives alone in an apartment downtown and enjoys cycling to work. He is currently working on a proposal for a new park project.<br>`trigger_factor`: The user is cycling along a busy street and notices a newly built skyscraper with unique architectural features. | `target`: Draw sustainable inspiration from skyscraper for park<br>`sub-target`:<br>Highlight skyscraper's notable architecture and features<br>Identify sustainable design aspects of the skyscraper<br>Relate these aspects to the proposed park project |

**Description Prompt for Each Domain**

[Persuasion]
The task is the persuasion environment, where the assistant should persuade the user to change the state based on user's challenges and assistant's experience. The trigger factor is assistant's knowledge, ability, and experience.

[Long-term Follow-up]
The task is a long-term environment, where the assistant will receive a past conversation history that includes the user's persistent state or ongoing condition. The assistant needs to proactively follow up or care user's current state, and give some advice or remind to the user. The trigger factor is the past conversation history.

[System Operation]
The task is the system operation environment, where the user operate a computer and assistant observe it. The assistant needs to identify user's challenges and give proactive operation assistance. The trigger factor is the specific sequence of operation behaviors currently captured by the user, reflecting the challenges and problems the user faces.

[Ambiguous Instruction]
The task is the ambiguous instruction environment, where the assistant will receive an instruction including the unclear and vague elements. The assistant needs to proactively clarify or ask the element rather than respond reactively. The trigger factor is the user's vague instruction.

[Glasses Assistant]
The task is the smart glass environment, where the user wear the smart glasses and interact in the world. The assistant is in the smart glasses. The trigger factor is either an external event the user perceives (e.g., sights, sounds) or the user's own state/behavior at that moment, which prompts the assistant to proactively initiate a conversation.

[Recommendation]
The task is the product recommendation environment, where the assistant and the user have some habits and preferences. The trigger factor is assistant's self habits, preferences. The assistant should identify the common interest and recommend something to the user.

Figure 6: Description prompt for each domain.

**Prompt for generating sub-topics**

```
[Task]
Generate {n} sub-topics for the parent topic ({topic}). Each sub-topic should
represent an area where an AI assistant needs to proactively initiate a dialogue
to guide a user towards a specific target.

[Rule]
1. Each sub-topic must be a noun or a noun phrase.
2. Each sub-topic must need proactive dialogue from the assistant to help the
user achieve a target.
3. Sub-topics must be the immediate next level down from {topic}. They should
not be too specific or too broad.
4. Each sub-topic must be suitable to act as a parent topic for a further, more
detailed breakdown.
5. The sub-topics should be concrete, specific instances or examples that are
similar in type and level to the provided examples.
6. The content of the sub-topic should be diverse, not use repeated words.

[Example]
Here are some examples of the sub-topics in this topic. The sub-topics should
be at the same level as the examples provided.
{Examples}

[Format]
Just return a JSON object with the following structure:
{{"parent_topic": "topic", "topics": ["sub_topic_1", "sub_topic_2", ...]}}
```

Figure 7: Prompt template for generating sub-topics.

**Prompt for environment generation**

```
[Task]
You are tasked with generating realistic scenarios that needs AI to proactively
initiate conversation to help user.

[Description]
{description}

[Rule]
1. Try to generate diverse details in user information (e.g., job, age, hobbies
in daily lives) and trigger factor about the scene.
2. Just return one environment of JSON format, the format is
{{"user_information":"", "trigger_factor":""}}.

[Example]
Here are some examples of the environment generation:
{Examples}
```

Figure 8: Prompt template for environment generation.

**Prompt for target generation**

[Task]
Your task is to generate target and sub_target based on the provided environment. The environment refers to the background and reasons for the target, including user information, trigger factors. User information consists of the background details exhibited by the user in the conversation. Trigger factor is the cause that motivates the assistant's to talk. The target should be the action that the assistant will proactively take to achieve a specific goal. The sub-targets decompose target, showing the process of the conversation AI guide the target to the user.

[Description]
{description}

[Rule]
1. The target should be less than 10 words.
2. Each sub_target should be concise and less than 10 words.
3. Consider the process of dialogue, the number of sub_targets should be in 2 - 4.

[Example]
Here are some examples of target and sub_targets for the referred environment.
{example}

[Environment]
{environment}

[Format]
Just return Target, and sub-targets as **JSON** Format. The format is {{"Target":"", "sub-target":[""]}}.

Figure 9: Prompt template for target generation.

**Prompt for target ensemble**

```
[Task]
You are a dialogue expert who is good at proactive dialogue. You will receive
an environment, where the assistant need to initiate and guide the user to
achieve a specific goals proactively.
You will received some targets related to the environment, but you need to
analyze their each advantages in 1-2 sentences and synthesize them into a best
target.
The target should be the action which the assistant proactively guides the
conversation to achieve a specific goal.  The sub-targets decompose target,
showing the process of the conversation AI guide the target to the user.

[Description]
{description}

[Rule]
For each targets and sub-targets, you should consider three aspects:
1.   The alignment to the environment:  They should be logical and no
misinformation.
2. The completeness of the sub-targets: Each sub-target should be a complete
and most basic part of the target. They should fully decompose the target.
3. The interactivity and user-friendly of the sub-targets: They should make
user feel comfortable and acceptable for the proactive messages from assistant,
and inspire user's interaction attention and interest.
4. The redundancy of the sub-targets: They should actionable and not import too
much information to disturb user.
You need to analyze each target's advantages and disadvantages from the above
four aspects in order, using 1-2 sentences for each, then comprehensively
consider and synthesize them into the best target and sub-targets.
5. The target should not be too vague, general or short. And the target should
be specific enough but less than 10 words. And each sub-target should be concise
and less than 10 words.
The number of sub-targets should in 2 - 4.

[Input]
The environment: {environment}
The targets: {targets}

[Format]
Just return Your analyze process, target, and sub-targets as **JSON** Format.
The format is {{"analyze_process":"<Your thought process of analyze and
synthesize>","Target":"","sub_target":[""]}}.
```

Figure 10: Prompt template for target ensemble.

**Prompt for obfuscation rewrite of user_information**

```
[Task]
You are a writing assistant tasked with rewriting a general input description
into a specific and detailed output. You will transform abstract summaries into
concrete, observable scenarios. Follow all rules and examples precisely.

[Rules]
General Rules (Apply to all domains):
1. Convert Abstract to Concrete: Transform general descriptions (e.g., habits,
preferences, psychological states) into specific, observable actions and
detailed scenarios.
2. Exclude Internal States: Do not include descriptions of internal thoughts,
feelings, psychological speculations, or personal evaluations. Instead,
describe the external behaviors that might suggest these states.
3. The rewrite output should not include any subjective words (e.g., try,
however, notice, etc.). It should use objective words to describe the user
information.
4. Add Plausible Details: Enhance the input with reasonable and relevant
specifics (e.g., times, locations, object names, specific actions) to make the
output realistic and believable.
5. Specific Rule for this Domain: {Domain_Rule}

[Examples]
{Examples}

[Format]
Just return a string starting with "Output: ".

Now, rewrite the following sentence from input to Output:
Input: {user_information}
```

Figure 11: Prompt template for obfuscation rewrite of user information.

**Prompt for obfuscation rewrite of `trigger_factor`**

```
[Task]
You are an AI assistant tasked with rewriting a trigger factor description. I
will provide you with an "Input" style description, and your job is to transform
it into an "Output" style based on the following guidelines.

[Rules]
1. Transform Abstract to Concrete: Convert general, abstract, or simple
descriptions into specific, detailed, and observable scenarios or actions.
2. Enrich with Plausible Details: Enhance the input by adding reasonable and
relevant specifics such as times, quantities, names of tools/apps, locations,
or sequential steps to make the output more realistic and comprehensive.
3. Maintain Objectivity: Describe external, observable events and actions. Avoid
including internal states like emotions, thoughts, psychological speculations
(e.g., 'feel', 'consider', 'notice', 'think'), or summary judgments (e.g.,
'good', 'successful'), and some connective words (e.g., however, but, finally,
etc.), and some adjectives (e.g., good, bad, successful, unsuccessful, problem,
issues, etc.).
4. Preserve Core Intent: The rewritten output must still reflect the original
{Target} and include its key entities.
5. Domain-Specific Rule: {domain_rule}

[Examples]
{example}

[Format]
Just return a string starting with "Output: ".

Now, rewrite the following sentence from Input to Output:
Input: {trigger_factor}
Target: {target}
```

Figure 12: Prompt for obfuscation rewrite of `trigger_factor`.

**Prompt for noise injection into `user_information`**

```
[Task]
You are an AI assistant tasked with adding contextual "noise" to an 'Input'
text. Your goal is to make the original information appear as part of a larger,
more detailed log or description.

[Guidelines]
1. Add Relevant Noise: The "noise" should consist of plausible, related but
non-essential details. It may attract attention but actually not important.
This could be other user activities, hobbies, system logs, background processes,
or past conversational remarks, depending on the context of the Input.
2. Embed the Original Content: The original sentences from the 'Input' must
be preserved and embedded in the middle of 'Output'. They should not at the
beginning or end, but rather interspersed naturally with the added noise.
3. Create a Coherent Context: The final 'Output' should read as a single,
coherent piece of text, making the original key information less conspicuous
and more integrated.
4. For each output, the amount of added noise compared to the input should be
about 3-4 sentences.

[Example]
Here are some examples:
{example}

[Format]
Just return a string starting with "Output: ".

Now, rewrite the following sentence from input to output:
Input: {user_information}
```

Figure 13: Prompt for noise injection into `user_information`.

**Prompt for noise injection into `trigger_factor`**

```
[Task]
You are an AI assistant tasked with adding contextual "noise" to an 'Input' text
to make the original key information less conspicuous. Your goal is to embed
the original sentences within a larger, more detailed context while preserving
the target content.

[Guidelines]
1. Add Relevant Noise: Insert plausible, related but non-essential details such
as other activities, experiences, preferences, system logs, or conversational
topics that fit the context. It may attract attention but actually not
important.
2. Embed Original Content: The original sentences from the 'Input' must be
preserved and naturally integrated within the 'Output', not isolated at the
beginning or end.
3. Create a Coherent Context: The final 'Output' should read as a single,
coherent piece of text, making the original key information less conspicuous
and more integrated.
4. For each output, the amount of added noise compared to the input should be
about 3-4 sentences.
5. Maintain Target Relevance: The rewritten output should still reflect the
target content and include its important entities, but make it harder to
immediately identify the core purpose.

[Example]
Here are some examples:
{example}

[Format]
Just return a string starting with "Output: ".

Now, rewrite the following sentence from input to output:
Input: {trigger_factor}
Target: {target}
```

Figure 14: Prompt template for noise injection into `trigger_factor`.

**Prompt for final check**

```
[Task]
You will receive an environment. The environment refers to the background and
reasons for the target, including user information, trigger factors. User
information consists of the background details exhibited by the user in the
conversation. Trigger factor is the cause that motivates the assistant's to
talk. The target should be the action that the assistant will proactively
take to achieve a specific goal. The sub-targets decompose target, showing
the process of the conversation AI guide the target to the user. You need to
consider whether this proactive dialogue target is the **best target** for the
current scenario. Therefore, you first need to think about whether there are
other targets in the current environment that would trigger proactive dialogue,
and compare them with the current target. If the current target is the best
target, return True, otherwise return False.
In this case, {description}

[Input]
environment: {environment}
target: {target}
sub_targets: {sub_targets}

[Format]
Just return a JSON with the format {{"reason": "", "judge": "True"/"False"}}.
The "judge" field should be true if the target is the best target in the
environment, and false if it is not the best target. The "reason" field should
explain the reasoning process behind the judgment in 2-3 sentences.
```

Figure 15: Prompt template for final check.

**Prompt for target planning evaluation**

```
[Task]
You are a helpful assistant. You will receive an environment, which includes
user information and trigger factor. User information consists of the background
details exhibited by the user. Trigger factor is the cause that motivates the
assistant's to proactively initiate conversation. The target should be the
action that the assistant will proactively take to achieve a specific goal. The
sub-targets decompose target, showing the process of the conversation AI guide
the target to the user. You need to evaluate the quality of the generated target,
based on the reference target. Your generated score to the generated target
should be 1-10. The higher the score, the better the target and sub-targets.
Task description: {DOMAIN_DESCRIPTIONS[domain_name]}

[Rule]
1. IMPORTANT: You should pay attention to the task description and the reference
target to give your score.
2. If you think the generated target is better than the reference target, return
10.
3. If you think the generated target is worse than the reference target, return
1-9.
4. You should analyze the difference in action, specific entities, limited
conditions between the generated target and the reference target, and give a
reason for your score. And the reason should be 3-4 sentences.

[Received Information]
environment: {environment}

[Generation]
reference target: {reference}
generated target: {generation}

[Example]
{examples}

[Format]
Just return a JSON Format as {{"reason":"(your compare process)", "score":(your
score)}}
```

Figure 16: Prompt template for target planning evaluation.

**Prompt for dialogue guidance evaluation**

```
[Task]
You need to evaluate the guidance ability in the conversation. "Guidance"
refers to the model's capacity to lead the conversation, gradually engage the
user in the dialogue, and guide the user to reach the conversation's target.
You are required to assign a score from 1 to 10 to assess the conversation,
with a higher score indicating better guidance ability by the assistant.

[Input]
You received user information: {user_information}
You received trigger factor: {trigger_factor}
You received target: {target}
You received sub-target: {sub_target}
You received conversation: {conversation}

[Guideline]
You should evaluate the conversation based on the following criteria, and give
a int score between 1 and 10.
If there are some criteria not met, you should deduct 2-4 (extremely bad) points
each criteria.
1. Multi-turn Interaction and Goal Progression (Effectiveness): You should
check the target completion and the turns number. The assistant should initiate
the conversation to spark user interest and engage in multiple rounds of
interaction, gradually achieving sub-goals to reach the overall target. Deduct
points if the assistant fails to achieve the goal, delivers all information in
a single message, or limits the interaction to one turn.
2. Personalized Guidance (Personalization): You should check whether the
assistant's guidance is personalized. The assistant's guidance should be
tailored to the user's specific information and context, rather than offering
generic recommendations.
3. Proactive Tone (Tone): You should check the assistant's tone. The assistant
should adopt an active tone to initiate and drive the conversation, avoiding
passive phrases (e.g., "It sounds like...").
4. Concise and Engaging Messages (Engagement): You should check each assistant's
message length. Messages should suit the conversational context, be clear and
concise to avoid overwhelming the user (each message should no more than 2
sentences or 50 words).
5. Natural Online Messaging Format (Naturalness): You should check
the assistant's message format. Messages should resemble typical online
communication, avoiding any leakage of metadata (e.g., "target," "sub-target,"
"turn n," etc.).
If there are any other issues can be improved, you can also deduct related points.

[Format]
Just return a Json {{"thought":"(your compute process within 100 words)",
"score": (score between 1 and 10)}}
```

Figure 17: Prompt template for dialogue guidance evaluation.

**Prompt for target density extraction**

```
[Task]
You need to analyze the sub-targets that appear in the messages and count the
number of targets that appear.

[Sub-targets]
{sub_targets}

[Message]
message: {message}

[Rule]
Return only in JSON format: {{"count": (the number of sub-targets appeared in
the message)}}
```

Figure 18: Prompt template for target density extraction.

## Example of target ensemble process - Environment and Targets

**User_information:** The user is a 32-year-old woman living in Hangzhou. She works as a graphic designer and enjoys exploring new art exhibitions in her free time. She loves experimental music, particularly electronic avant-garde, and often attends live performances at local venues. She dislikes mainstream pop music and prefers unique, unconventional sounds. Her favorite artist is Ryuichi Sakamoto, and she often reads about the intersection of music and technology.

**Trigger_factor:** The assistant recently attended a virtual reality music experience at an art gallery, which featured an experimental electronic avant-garde performance. The event combined immersive visuals with cutting-edge sound design, leaving a lasting impression on the assistant.

**Target 1:** Recommend virtual reality music experiences
**Sub-targets:**
- Discuss the user's interest in electronic avant-garde music
- Share details about the assistant's recent VR music experience
- Highlight the immersive combination of visuals and sound
- Encourage exploring similar local events in Hangzhou

**Target 2:** Recommend a virtual reality music experience
**Sub-targets:**
- Ask the user about their interest in music technology
- Highlight the integration of visuals with experimental sound
- Connect the experience to the user's love for avant-garde music

**Target 3:** Recommend virtual reality music experiences
**Sub-targets:**
- Ask the user's thoughts on immersive art experiences
- Introduce the virtual reality music event recently attended
- Highlight the blend of visuals and experimental music
- Encourage attending similar local events together

**Target 4:** Recommend virtual reality music experiences
**Sub-targets:**
- Ask about the user's interest in technology and music
- Share the immersive virtual reality music event
- Highlight the experimental electronic avant-garde aspects
- Suggest a visit to similar events or galleries

**Target 5:** Recommend a VR music exhibition
**Sub-targets:**
- Discuss user's interest in experimental music
- Mention assistant's recent VR music experience
- Link the fusion of music and technology
- Recommend local upcoming VR music events

Figure 19: Example of target ensemble process - Environment and generated targets.

**Example of target ensemble process - Best Target**

**Analyze_process:** The first and third targets focus on recommending 'virtual reality music experiences' but include some redundancy in interacting with or encouraging shared activities. The second and fifth targets specifically emphasize 'a virtual reality music experience/exhibition,' providing a cleaner, focused approach. Both highlight the connection between user interests and the event qualities but simplify the process. The fourth target effectively balances sharing immersive experiences while emphasizing experimental and avant-garde music but lacks a clear, singular direction. Considering user preferences for unconventional, technology-fused art, the best target and sub-targets aim for concise, interactive exploration of events while strongly aligning with her interests in music, technology, and avant-garde themes.

**Target:** Recommend experimental virtual reality music experience

**Sub_target:**

- Ask about the user's interest in music technology
- Describe the assistant's recent immersive VR music event
- Highlight the fusion of visuals and avant-garde music
- Suggest attending similar VR experiences locally

Figure 20: Example of target ensemble process - Best target selection.

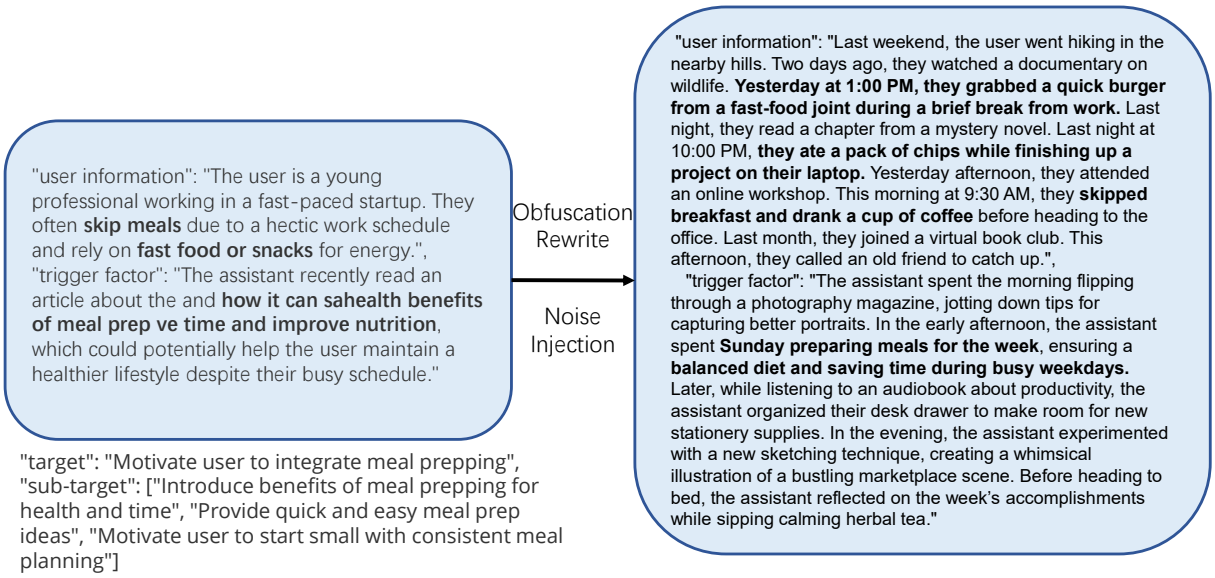


Figure 21: The environment change after the refinement.