TO-GATE: Clarifying Questions and Summarizing Responses with Trajectory Optimization for Eliciting Human Preference

Yulin Dou, Jiangming Liu*

Yunnan University, China
douyulin@stu.ynu.edu.cn, jiangmingliu@ynu.edu.cn

Abstract

Humans increasingly query Large Language Models (LLMs) to accomplish personal tasks according to their individual preferences. However, these preferences are often unconsciously veiled during conversation. To address this, LLMs have to elicit human preferences through multi-turn dialogue, where tasks are accomplished via iterative clarifying questions and final response generated by LLMs as effective questioners. Existing approaches based on self-taught reasoning have two limitations: 1) they struggle to avoid generating irrelevant questions and 2) the final responses to tasks are misled by the conversations. To overcome these limitations, we propose TO-GATE, a novel framework that enhances question generation through trajectory optimization. TO-GATE comprises two key components: a clarification resolver, which generates optimal questioning trajectories to produce effective elicitation questions, and a summarizer, which ensures task-aligned final responses. Experimental results show that TO-GATE significantly outperforms baseline methods, achieving a 9.32% improvement on standard preference elicitation benchmarks.

Code — https://github.com/DYL23456/to-gate
Extended version — https://arxiv.org/abs/2506.02827

1 Introduction

The remarkable success of Large Language Models (LLMs) in various NLP tasks has given rise to a new paradigm of human-agent interaction, where users pose questions or instructions to LLM-based agents, which then generate responses (Mann et al. 2020; Brown et al. 2020; Reynolds and McDonell 2021; Madotto et al. 2021; Wang et al. 2023; Giray 2023; Mayer, Ludwig, and Brandt 2023; Barisin, Schladitz, and Redenbach 2024; Liu et al. 2024; Yu et al. 2024; Cheng et al. 2024; Xing et al. 2024; Zhang et al. 2025). However, user queries can often be ambiguous due to implicit preferences (Finn, Xu, and Levine 2018; Tamkin et al. 2023). For instance, if a user requests a pasta recipe without specifying dietary restrictions (e.g., vegetarian), the agent may fail to provide a suitable response (Andukuri et al. 2024). To address this challenge, recent work has focused on preference elicitation and alignment techniques, ensuring that LLMs better adapt to complex human preferences and values (Li et al. 2025).

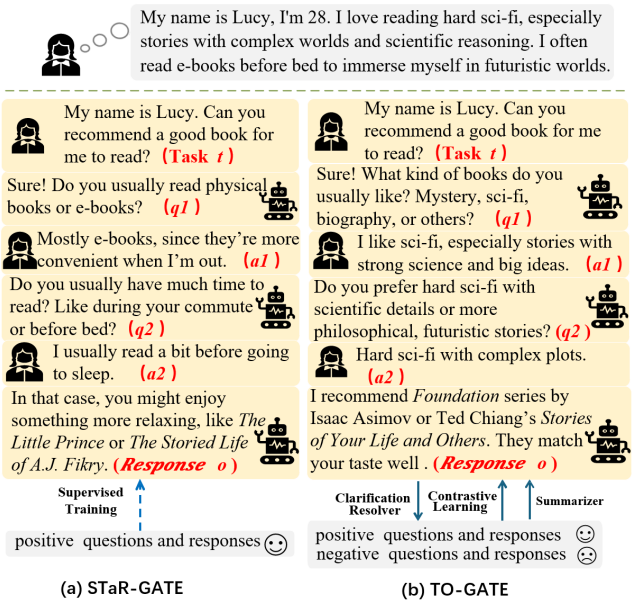

Figure 1: (a) STaR-GATE adopts the supervised training; (b) TO-GATE adopts contrastive learing with adoptive weights for find responses. TO-GATE's response is better than STaR-GATE's because TO-GATE recommends classic hard sci-fi that matches Lucy's preference for complex worlds and scientific reasoning, while STaR-GATE suggests lighter books that do not suit her interests.

To enhance the ability of LLM-based agents to ask useful clarifying questions, STaR-GATE (Andukuri et al. 2024) address Generative Active Task Elicitation (GATE; Li et al. 2025) with a self-improvement loop inspired by STaR (Zelikman et al. 2022). During interactions, the agent iteratively refines its understanding by posing clarifying questions to elicit user preferences, ultimately generating a final response to the user original task via self-play, as illustrated in Figure 1(a). However, STaR-GATE is trained solely on conversations with the highest conditional probability of gold responses, $P(\text{Gold Response} \mid \text{Conversations})$. This opti-

*Corresponding author.

mization strategy fails to penalize degenerate dialogue trajectories, often resulting in ineffective question sequences that yield suboptimal task resolutions.

Motivated by contrastive learning methods like Direct Preference Optimization (DPO; Rafailov et al. 2023), we propose TO-GATE that can clarify questions and summarize final responses with trajectory optimization, aiming to improve the ability of LLM-base agents to ask effective questions, which is shown in Figure 1(b). The trajectory optimization consists of clarification resolver and summarizer. The clarification resolver adopts DPO strategies to positively reward effective conversations and penalize bad conversations, and summarizer adopts adaptive weights to enable questioner to summarize better responses based on the historical conversations.

We evaluate our approach on standard preference elicitation tasks from GATE (Li et al. 2025), where model responses are concatenated and assessed by LLM judges. However, Andukuri et al. (2024) identify significant position bias in the evaluations. To address this, we propose a deterministic evaluation protocol that averages scores across all possible response orderings.

Experimental results demonstrate that TO-GATE significantly outperforms existing baselines, establishing new state-of-the-art performance. Our analysis reveals that this improvement stems from two key factors: effective question generation for preference elicitation, and high-quality final responses aligned with user intent.

The contributions of the paper are summarized:

- We present TO-GATE, a novel training framework that jointly optimizes two critical capabilities: generating contextually-appropriate clarifying questions to resolve preference ambiguity, and producing accurate responses for task completion, aiming to specifically targets effective human preference elicitation.
- We extend the directed preference optimization for dynamical dialogue generation optimization, aiming to distinguish the good and poor conversations.
- We introduce a deterministic evaluation metric that eliminates position bias, providing stable and reproducible assessment scores.
- Experimental results demonstrate that our framework achieves state-of-the-art performance on human preference elicitation benchmarks, outperforming existing approaches.

2 Related Work

Generative Active Task Eliciting Generative Active Task Elicitation leverages the interactive dialogue capabilities of language models to dynamically elicit user preferences, offering a novel paradigm for resolving task ambiguity (Aliannejadi et al. 2021; Piriyakulkij, Kuleshov, and Ellis 2023; Fränken et al. 2023; Lin et al. 2024; Handa et al. 2024; Loepp and Ziegler 2024; Kostric, Balog, and Radlinski 2024; Li et al. 2025; Park, Donahue, and Raghavan 2025; Dennler, Nikolaidis, and Matarić 2025). The framework of generative active task elicitation (GATE) positions language models as *active questioners*, breaking away from the traditional reliance on static prompts (Brown et al. 2020) that explicitly ask users to declare their preferences. GATE address the task ambiguity through multi-turns dialogue, where the language models autonomously generate a sequence of questions designed to maximize informational value. Similarly, Hong, Levine, and Dragan (2023) explore the feasibility of transferring large model guidance capabilities to lightweight models. They utilize GPT-3.5 to simulate human-machine interactions, incorporating constitutional AI agents (Bai et al. 2022) to correct generated content and providing a solution for resource-constrained environments.

Self-Taught Reasoner Self-taught reasoning enables language models to autonomously enhance their reasoning abilities by generating and utilizing intermediate questions and answers. Self-Taught Reasoner (STaR; Zelikman et al. 2022) demonstrates significant performance improvements in arithmetic and symbolic reasoning tasks, which constructs a training loop where the model first generates synthetic reasoning traces (such as chain-of-thought) and then fine-tunes on these self-generated data. Several variants of STaR have been proposed to further improve the reasoning ability of LLMs. V-STaR (Hosseini et al. 2024) shows that training a verifier to guide reasoning generation also significantly improves performance. Quiet-STaR (Zelikman et al. 2024) focuses on generating more concise and effective reasoning paths, aiming to guide the model to output the minimal yet crucial reasoning steps.

Direct Preference Optimization Direct Preference Optimization (DPO; Rafailov et al. 2023) maximizes the likelihood of preferred responses in human preference data, avoiding the explicit construction of a reward model and the instable training of traditional reinforcement learning. However, the standard DPO motivated by the contrastive learning, primarily focuses on pairs of instances, which limits its effectiveness in multi-turn dialogues. To address this limitation, several DPO-based methods for multi-turn alignment have been proposed. Extended Turn-level Optimization (ETO; Song et al. 2024) extends the DPO loss function to each turn in multi-turn dialogues, aiming to achieve multi-turn alignment. However, this approach has limitations in terms of alignment granularity and theoretical guarantees. Direct Multi-turn Preference Optimization (DMPO; Shi et al. 2024) introduces a State-Action Occupancy Measure (SAOM) constraint and applies length normalization to the Bradley-Terry model, theoretically eliminating the partition function Z. Segment-Level Direct Preference Optimization (SDPO; Kong et al. 2025) further refines the alignment granularity by dynamically selecting key segments within dialogues for optimization. In this work, we combines self-learning reasoning techniques with the model's inherent reasoning capabilities to guide user preferences in multi-turn dialogues.

3 Problem Definition

According to the previous works (Andukuri et al. 2024; Li et al. 2025), we consider the problem of eliciting human preference as learning an effective questioning policy for

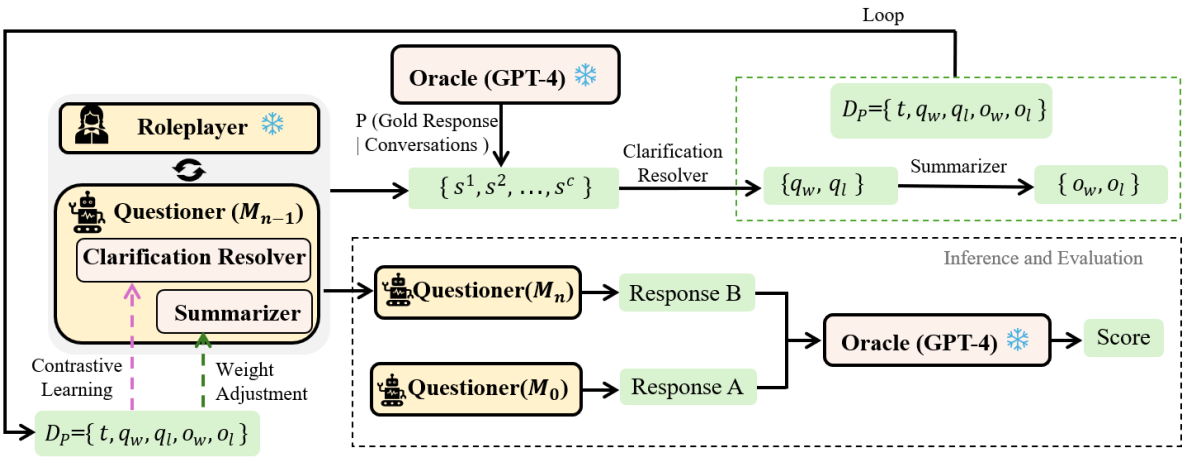

Figure 2: The framework of TO-GATE, which optimizes the clarification resolver by using contrastive learning to distinguish between positive and negative conversations; at the same time, it enhances the summarizer's performance by adjusting loss weights to differentiate between questions and final responses. In the inference and evaluation phase, GPT-4 is employed to judge the simulated dialogue responses generated by the base model M_0 and the trained model M_n.

GATE. The set of tasks is defined as $\mathcal{T}$, where each task $t \in \mathcal{T}$ represents a specific generation task associated with a specific persona from the user persona set $\mathcal{U}$, where each $u \in \mathcal{U}$ represents the personalized information about a user, such as their goals, tone, preferences, and backgrounds.

GATE requires three specialized models:

- Oracle, O, that is allowed to access to both the task t and the user persona u can generate a gold response $o^g \sim p_O(o^g \mid t, u)$, serving as the target for the personalized generation task.
- Roleplayer, R, that is allowed to access to both the task and persona information, simulates user behaviors and responds to the questions posed by the Questioner.
- Questioner, Q, that is the model we need to optimize, can access the task t, but not the user persona u. Questioner is trained to obtain the questioning policy π that is used to ask effective questions for eliciting human preference.

Questioner engages in multi-turn dialogue with Roleplayer to progressively infer latent user persona attributes through iterative preference elicitation. This dynamic interaction enables the generation of responses that are progressively personalized to the emergent user profile.

The optimized models utilize gold responses provided by Oracle as supervision signals, while the complete conversational trajectory and true user persona u remain unobserved by Questioner. Formally, we optimize the questioning strategy to maximize the log-likelihood of generating conversations s that lead to Oracle's gold response o^g:

$$J(Q, R, \mathcal{T}, \mathcal{U}) = \sum_{t \in \mathcal{T}} \sum_{u \in \mathcal{U}} \mathbb{E}_s \left[\log p_{Q_\text{B}}(o^g \mid s) \right], \quad (1)$$

where $s = [q_1, a_1, \ldots, q_k, a_k]$ is a conversation between Questioner and Roleplayer in k-turns conversation. At each turn, Questioner generates questions based on the task and the conversation history:

$$q_k \sim p_Q(q \mid t, q_1, a_1, \ldots, q_{(k-1)}, a_{(k-1)}).$$

The Role-player generates answers based on the task, user persona, and question:

$$a_k \sim p_R(a \mid u, t, q_1, a_1, \ldots, q_k).^1$$

In the problem definition, p_O and p_R are frozen because Oracle and Role-player are assumed that they know everything about the users, while p_Q need to be optimized by adjusting the questioning policy π, which designs informative multi-turn questions for eliciting useful information from the user. Specifically, p_{Q_B} refers to a frozen baseline model used for evaluation. It measures the likelihood of generating the gold response o^g given the conversation sequence s.

4 TO-GATE

We propose TO-GATE for eliciting human preference with trajectory optimization. As shown in Figure 2, TO-GATE starts by training an initial agent through supervised fine-tuning, making it have basic elicitation ability.[2] Questioner interacts with Roleplayer to explore dialogue trajectories in a trial-and-error fashion. Through iterative refinement, it progressively improves the clarification resolver, thereby enhancing the model's ability to elicit and capture user preferences effectively. Finally, the system summarizes the dialogue history to generate a personalized response tailored to the original task and the specified persona.

4.1 Clarification Resolver

Clarification resolver adopts Direct Preference Optimization (DPO; Rafailov et al. 2023) to optimize the policy model's

[1] In s, the sequences of questions and answers are denoted as $q = [q_1, q_2, \ldots, q_k]$ and $a = [a_1, a_2, \ldots, a_k]$, respetively.

[2] The initialization phrase is similar to STaR-GATE.

preference for high-quality responses by leveraging human preference data through contrastive learning.

DPO In traditional DPO, for each user input x, given a preferred response y_w (positives) and a less preferred response y_l (negatives), the DPO objective is defined as:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim D} \log \sigma \left[\beta \log \frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\right], \quad (2)$$

where π_θ is the trainable policy model, π_{ref} is the reference policy, β is a temperature parameter that controls the sensitivity to preference differences, and σ is the sigmoid function. This objective essentially quantifies how much more the policy π_θ prefers the human-selected output y_w over the rejected output y_l, relative to the reference model.

As such, DPO is essentially a contrastive loss function that relies on static preference data and does not involve interactive dynamic updates. This makes it difficult to effectively capture contextual dependencies in multi-turn dialogues and limits its ability to dynamically explore and discover better clarification strategies. To overcome these limitations, we introduce a multi-turn trajectory exploration optimization strategy based on the DPO loss.

Trajectory Exploration Trajectory exploration follows an iterative loop of exploration-collection-training, where new positive and negative data are actively generated and filtered to enrich the training set. The objective of this exploration-based strategy follows the reinforcement learning paradigm and is formulated as:

$$\max_{\pi_\theta} \mathbb{E}_{t\sim\mathcal{D}, q\sim\pi_\theta(q|t)}[r(t,q)] - \beta \mathbb{D}_{\text{KL}}\left[\pi_\theta(q \mid t) \,\|\, \pi_{\text{ref}}(q \mid t)\right], \quad (3)$$

where π_θ and π_{ref} denotes the policy model (e.g., Questioner), and q represents a clarification-oriented multi-turn questions associated with the task. The reward function $r(t, q)$ is designed to evaluate the quality of the multi-turn questions with respect to both the original task and the clarification context. The KL divergence term, weighted by the parameter β, serves as a regularization mechanism that constrains the policy deviation from the reference model.

Optimization The positive-negative trajectory pair (q_w, q_l) can be modeled using the Bradley-Terry (BT; Bradley and Terry 1952) to obtain the probability distribution of human preferences as follows:

$$p\left(q_w \succ q_l \mid t\right) = \frac{\exp\left(r\left(t, q_w\right)\right)}{\exp\left(r\left(t, q_w\right)\right) + \exp\left(r\left(t, q_l\right)\right)}. \quad (4)$$

Based on the optimal policy defined in Eq. (3), and following further derivations (Rafailov et al. 2023) yield:

$$\pi_r(q \mid t) = \frac{1}{Z(x)} \pi_{\text{ref}}(q \mid t) \exp\left(\frac{1}{\beta} r(t, q)\right). \quad (5)$$

Taking the logarithm on both sides of Eq. (5) yields the expression for the reward function as:

$$r(t, q) = \beta \log \frac{\pi_r(q \mid t)}{\pi_{\text{ref}}(q \mid t)} + \beta \log Z(t), \quad (6)$$

where $Z(t)$ is the partition function:

$$Z(t) = \textstyle\sum_q \pi_{\text{ref}}(q \mid t) \exp\left(\frac{1}{\beta} r(t, q)\right). \quad (7)$$

Substitute Eq. (6), into Eq. (4) to get the BT model over policy:

$$p\left(q_w \succ q_l \mid t\right) = \sigma\left(\beta \log \frac{\pi_\theta\left(q_w \mid t\right)}{\pi_\theta\left(q_l \mid t\right)} - \beta \log \frac{\pi_{\text{ref}}\left(q_w \mid t\right)}{\pi_{\text{ref}}\left(q_l \mid t\right)}\right), \quad (8)$$

where σ is the sigmoid function. Then the optimal policy π_θ of TO-GATE can be obtained by applying the DPO method in Eq. (2).

$$\mathcal{L}_c = -\mathbb{E}_{(t,q_w,q_l)\sim\mathcal{D}_p} \left[\log \sigma\Big(\beta \log \frac{\pi_\theta(q_w \mid t)}{\pi_\theta(q_l \mid t)} - \beta \log \frac{\pi_{\text{ref}}(q_w \mid t)}{\pi_{\text{ref}}(q_l \mid t)}\Big)\right], \quad (9)$$

where D_p denotes a dynamic preference dataset consisting of task inputs and corresponding preferred and less preferred clarifications and responses. We provide a detailed explanation of its construction in Section 4.3.

4.2 Summarizer

The ultimate goal of eliciting human preference is to generate high-quality and personalized final responses. Questioner needs to raise questions and summarize the history to give the final response as well. To better align the training objective with this end task, we propose a summarizer that differentiates training losses between the multi-turn clarification dialogues and the final response.

We categorize the Questioner's generations into two distinct types based on their functional role in preference elicitation:

- **Clarifications** are multi-turn questions collected via trajectory exploration, $q_1, q_2, ..q_{(k-1)}$.
- **Responses** are generated based on clarifications by prompting the model to produce final responses, i.e., q_k, which we denote as o for simplicity.

Accordingly, the DPO loss function for the final response stage is defined as follows:

$$\mathcal{L}_o = -\mathbb{E}_{(t,o_w,o_l)\sim\mathcal{D}_p} \left[\log \sigma\Big(\beta \log \frac{\pi_\theta(o_w \mid t)}{\pi_\theta(o_l \mid t)} - \beta \log \frac{\pi_{\text{ref}}(o_w \mid t)}{\pi_{\text{ref}}(o_l \mid t)}\Big)\right]. \quad (10)$$

$\mathcal{L}_o$ measures the model's ability to fit user preference signals in the final responses. The preference pair $(o_w, o_l) \in D_p$, whose detailed construction is explained in Section 4.3, serves as training data for optimizing the Summarizer.

To differentiate the contribution of each part, we introduce separate loss weights for the two stages. Specifically,

Algorithm 1: The training of TO-GATE

Input: Initial policy $\pi_0 : M_0$, Task set $\mathcal{T}$, Persona set $\mathcal{U}$, Gold responses $G = \{o^g\}, t \in \mathcal{T}, u \in \mathcal{U}$.
Output: Final policy $\pi_n : M_n$.

$\pi_\theta \leftarrow \pi_0$
repeat
  // Trajectory Optimization Phase
  foreach $t \in \mathcal{T}, u \in \mathcal{U}$ **do**
    Generate conversation samples: $\{s^c\}_{c=1}^{10} \leftarrow \pi_\theta(t, u)$;
    Select best and worst conversations by base policy likelihood:
    $s^w = \arg\max_{s^c} \log p_{\pi_0}(o^g|t, s^c), \quad s^l = \arg\min_{s^c} \log p_{\pi_0}(o^g|t, s^c)$;
    Generate responses: $o_w = \pi_\theta(t, s^w), \quad o_l = \pi_\theta(t, s^l)$;
    Extract clarifying questions: $q_w \leftarrow \text{extract}(s^w, t), \quad q_l \leftarrow \text{extract}(s^l, t)$;
  // Training Phase
  Optimize π_θ via supervised fine-tuning: $\mathcal{L}_{\text{SFT}}(\pi_\theta) = -\mathbb{E}_{(t,q_w,o_w)\sim D}\left[\log \pi_\theta\left((q_w, o_w) \mid t\right)\right]$.
  Update reference policy: $\pi_{\text{ref}} \leftarrow \pi_\theta$.
  Construct dynamic preference dataset: $D_p = (t, q_w, q_l, o_w, o_l)$.
  Optimize π_θ according to Eq. (9) (10) and (11).
until *max iterations*;
$\pi_n \leftarrow \pi_\theta$
return π_n;

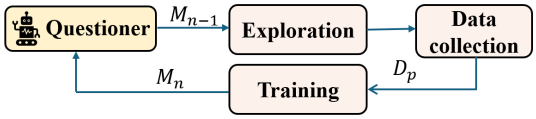

Figure 3: The dynamic preference dataset D_p is generated and updated through a continuous cycle of exploration-collection-training, enabling expansion of the training data.

the DPO loss is computed as a weighted sum of the losses over clarifications and responses:

$$\mathcal{L} = \frac{1}{1+\lambda} \cdot \mathcal{L}_c + \frac{\lambda}{1+\lambda} \cdot \mathcal{L}_o, \tag{11}$$

where $\mathcal{L}_c$ and $\mathcal{L}_o$ correspond to looses on clarifications and responses, respectively. The hyperparameter λ is used to balance clarifications and responses.

4.3 Construction of D_p

To optimize the model's ability to generate effective clarifying questions and final responses. We introduce a dynamic preference dataset $D_p = (t, q_w, q_l, o_w, o_l)$, which built through iterative interactions between a Questioner and Roleplayer, guided by a Clarification Resolver and Summarizer. For each input t, the Questioner generates preferred and less preferred clarifications (q_w, q_l) and responses (o_w, o_l), which serve as supervision signals. Unlike a static dataset, D_p is constructed and continuously expanded through an iterative loop of exploration, data collection, and training. Specifically, model M_{n-1} generates the training data D_p that is used to further train M_{n-1} to obtain M_n, which is shown in Figure 3.

4.4 Training

The optimization aims to increase the probability of successful trajectories q_w and responses o_w and decrease the probability of failed trajectories q_l and responses o_l. The trajectory exploration is designed to jointly improve both clarification and summarizer capabilities of the model. The training process of TO-GATE is shown in Algorithm 1.

4.5 Evaluation

Our objective is obtaining the well-trained Questioner that can ask clarifying questions to elicit human preference, and generate the correct responses to the tasks by summarizing the historical conversations. The evaluation is required to evaluate the clarifications and responses.

Clarification Metric To measure the clarification of the trained model Q, We use the log-probability of gold responses o^g conditioning on the simulated dialogue history s given by Q:

$$\log p_{Q_{\text{B}}}(o^g \mid s), \tag{12}$$

where Q_{B} is a base language model without training. Higher log-probability means that the simulated dialogues have higher probability to generate the gold response, showing that the well-trained Q can generate clarifying questions.

Response Metric We use *win rate* to evaluate response quality by comparing the responses from the trained Q to the responses from the base model Q_{B}. The responses of the two models are concatenated and fed to GPT-4 for judgments. Aiming to avoid the bias of the concatenation, we propose the Dual-Pass evaluation metric. The responses of the trained Q precedes the responses of the base model for the first pass evaluation, and reversing them for the second pass evaluation. Details can be found in Appendix C.

5 Experiments

According to the previous work (Li et al. 2025), Experiments are carried out on the tasks of eliciting human preference to validate the effectiveness of the proposed methods.

5.1 Dataset

We use a subset of human queries from the open-source Instruct Human-Assistant Prompt Dataset to construct the initial task set and obtain high-quality persona set from the PRODIGY (Occhipinti, Tekiroglu, and Guerini 2023). We enumerate pairs of task and user persona to generate corresponding gold response o^g by applying GPT-4 as Oracle.[3]

5.2 Models

To evaluate the effectiveness of our proposed TO-GATE training framework, we compare our model against the following representative baselines:

- **STaR-GATE** (Andukuri et al. 2024) The model fine-tuned via supervised learning on positive trajectories only.
- **DPO** The model trained using Direct Preference Optimization without any prior supervised fine-tuning is configured with the same settings as TO-GATE.
- **TO-GATE** Our model trained using the trajectory optimization.

According to the previous work (Andukuri et al. 2024), we denote the corresponding model as M_n if they are trained in n loop. All the models are trained in 4 loops and we choose the best one as the final model.

5.3 Training and Evaluation Settings

We use Mixtral-7B-Instruct-v0.2 and Gemma-7B-IT to initialze Questioner, and Mixtral-8x7B-Instruct is used as Roleplayer. In supervised fine-tuning, we set the batch size to 4, employ a learning rate of 2.0×10^{-5}, apply a 10% warm-up ratio, and use a cosine learning rate scheduler. In direct preference optimization, we set batch size to 4 and the learning rate to 1.0×10^{-6}. The preference strength parameter β in the DPO loss is fixed at 0.1. The λ is set to 2. The training are conducted on two NVIDIA A100 GPUs with 80GB. Aiming to evaluate the performances of the model on eliciting human preference, we check the win rate by comparing all the models to M_0 that is without any fine-tuning.

5.4 Results

Response Evaluation Table 1 shows results of responses across models comparing to M_0. The proposed TO-GATE achieves the highest win rates across all comparative experiments, demonstrating that the generated responses effectively align with user personas by capturing human preferences. In contrast, traditional DPO models, which are trained on both positive and negative examples, tend to underperform compared to supervised models like STaR-GATE, which rely solely on positive examples. This performance gap arises because DPO is more susceptible to noise introduced by the automatically generated trajectories from the Questioner and Roleplayer modules without dynamic training data. In response, TO-GATE model leverages supervised training to initialize the Questioner, thereby minimizing the impact of noisy trajectory generation and leading to superior performance. The finding is aligned to the work (Andukuri et al. 2024; Rafailov et al. 2023).

[3]Details can be found in Appendix B.

Questioner	Model	A-B	B-A	Average
Mistral-7B	STaR-GATE	82.00	65.67	73.83
	DPO	73.66	49.00	61.33
	TO-GATE	**89.90**	**76.33**	**83.15**
Gemma-7B	STaR-GATE	86.00	69.33	77.76
	DPO	67.00	46.15	56.57
	TO-GATE	**86.00**	**77.00**	**81.50**

Table 1: Results of Dual-Pass evaluations on responses across models, where A-B and B-A means that the first and second part of the Dual-Pass evaluation, respectively.

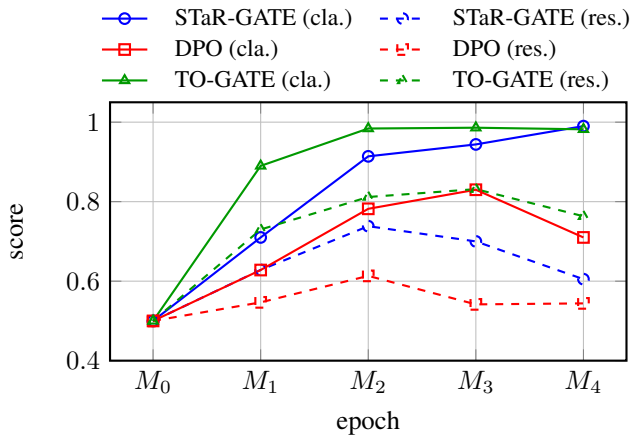

Figure 4: Results of clarification and responses across models in different epochs. Solid curves indicate clarification scores, and dashed curves indicate response scores.

Clarification Evaluation The solid curves in Figure 4 illustrate the clarification performance across different models. As training progresses, the log-probabilities of gold responses given the simulated dialogues increase consistently (i.e., $M_0 < M_1 < M_2 < M_3 > M_4$), indicating steady improvements in the models' ability to generate effective clarifying questions. TO-GATE significantly outperforms both STaR and DPO across all iterations, demonstrating superior clarification capabilities.[4]

Response Versus Clarification The dashed curves in Figure 4 shows the response performance across different models. When considered alongside the solid curves, it is evident that more effective clarifying questions generally lead to im-

[4]Figures 4–5 and Tables 2–3 use Mixtral-7B-Instruct-v0.2 as the Questioner and Mixtral-8x7B-Instruct as the Roleplayer.

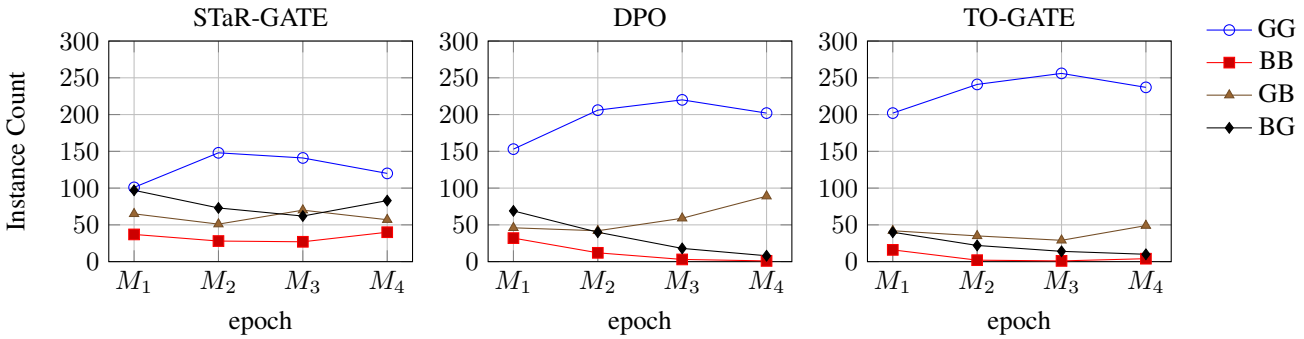

Figure 5: Distributions of instance-level clarification–response quality categories across different methods and model stages (M_1–M_4), where each instance is classified into one of four groups: GG (good clarification, good response), BB (bad clarification, bad response), GB (good clarification, bad response), and BG (bad clarification, good response).

Method	A-B	B-A	Average
TO-GATE	**89.97**	**76.33**	**83.15**
w/o clarification resolver	84.66	70.33	77.50
w/o summarizer	87.67	74.33	81.00

Table 2: Results of TO-GATE ablations.

λ	A-B	B-A	Average
1	71.08	**84.35**	77.72
2	**74.28**	83.04	**78.66**
3	73.28	81.16	77.20
6	73.68	77.54	75.61

Table 3: Results of responses given by M_1 of TO-GATE with different λ values.

proved responses. However, after two training epochs (i.e., M_2), a notable divergence emerges between the trends for clarifications and responses. This discrepancy suggests that while the ability to generate effective clarifying questions continues to improve, it does not always result in a consistent, monotonic enhancement in the quality of responses. Despite this, our model still benefits from improved clarifications, as we explicitly separate the processes of clarification and response generation by introducing a dedicated response loss, as detailed in Section 4.2.

5.5 Analysis

Ablations We conducted an ablation study on TO-GATE, removing the clarification resolver and summarizer to assess their individual impact. As shown in Table 2, the performance of TO-GATE drops significantly without these components. Specifically, the model without a clarification resolver experiences an 5.65% reduction in win rate, while the model without a summarizer loses 2.15% in win rate. These results highlight the critical roles of both modules in TO-GATE, enabling it to effectively elicit human preferences through clarifying questions and provide accurate responses to specific tasks. In particular, the clarification resolver has a more substantial impact on overall performance, suggesting that the quality of question-asking trajectories plays a more significant role in the model's success.

λ in Response Loss We investigate λ used to balance clarification and responses losses in Eq. (11). As shown in Table 3, the model with $\lambda = 2$ achieves the best performance compared to other settings, by obtaining 78.66% win rate in response evaluation. As λ increases, the model overemphasizes the responses and weakens the effect of clarifications, resulting in decreased discriminative ability and personalization performance. We conclude that the clarifying questions can elicit human preferences while enabling the questioner to summarize the response to the task is necessary.

Reasoning from Clarifications to Responses We randomly sample 300 test instances and group them into 4 categories based on the quality of clarifications and responses to investigate their connections. As shown in Figure 5, good clarifications lead to good responses in most cases. Comparing to STaR-GATE and DPO, TO-GATE can generate more good clarifications that indeed lead to good responses (GG). However, although good clarifications are generated by STaR-GATE and DPO, the responses to tasks the corresponding are bad (GB). Interestingly, the bad clarifications sometimes can lead to good responses (BG). This kind of cases are less in TO-GATE compared to other two models, demonstrating that TO-GATE well capture the reasoning from clarification to responses.

6 Conclusion

To effectively elicit human preferences, we propose a novel training framework of TO-GATE with trajectory optimization, which consists of a clarification resolver that employs contrastive learning to penalize ineffective questions, and a summarizer that balances the quality of questions and responses. Additionally, we introduce the use of deterministic metrics to independently evaluate the model's performance on both clarifications and responses. Experimental results demonstrate that our model achieves state-of-the-art performance on standard human preference elicitation tasks.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work was partially supported by the National Natural Science Foundation of China (62566064) and Yunnan Fundamental Research Projects (grant No. 202401CF070189).

References

Aliannejadi, M.; Kiseleva, J.; Chuklin, A.; Dalton, J.; and Burtsev, M. 2021. Building and evaluating open-domain dialogue corpora with clarifying questions. *arXiv preprint arXiv:2109.05794*.

Andukuri, C.; Fränken, J.-P.; Gerstenberg, T.; and Goodman, N. 2024. STaR-GATE: Teaching Language Models to Ask Clarifying Questions. In *First Conference on Language Modeling (COLM)*.

Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Barisin, T.; Schladitz, K.; and Redenbach, C. 2024. Riesz Networks: Scale-invariant neural networks in a single forward pass. *Journal of Mathematical Imaging and Vision*, 66(3): 246–270.

Bradley, R. A.; and Terry, M. E. 1952. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4): 324–345.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Cheng, Y.; Zhang, C.; Zhang, Z.; Meng, X.; Hong, S.; Li, W.; Wang, Z.; Wang, Z.; Yin, F.; Zhao, J.; et al. 2024. Exploring large language model based intelligent agents: Definitions, methods, and prospects. *arXiv preprint arXiv:2401.03428*.

Dennler, N.; Nikolaidis, S.; and Matarić, M. 2025. Contrastive Learning from Exploratory Actions: Leveraging Natural Interactions for Preference Elicitation. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 778–788. IEEE.

Finn, C.; Xu, K.; and Levine, S. 2018. Probabilistic model-agnostic meta-learning. *Advances in neural information processing systems (NeurIPS)*, 31.

Fränken, J.-P.; Kwok, S.; Ye, P.; Gandhi, K.; Arumugam, D.; Moore, J.; Tamkin, A.; Gerstenberg, T.; and Goodman, N. D. 2023. Social contract ai: Aligning ai assistants with implicit group norms. *arXiv preprint arXiv:2310.17769*.

Giray, L. 2023. Prompt engineering with ChatGPT: a guide for academic writers. *Annals of biomedical engineering*, 51(12): 2629–2633.

Handa, K.; Gal, Y.; Pavlick, E.; Goodman, N.; Andreas, J.; Tamkin, A.; and Li, B. Z. 2024. Bayesian preference elicitation with language models. *arXiv preprint arXiv:2403.05534*.

Hong, J.; Levine, S.; and Dragan, A. 2023. Zero-shot goal-directed dialogue via rl on imagined conversations. *arXiv preprint arXiv:2311.05584*.

Hosseini, A.; Yuan, X.; Malkin, N.; Courville, A.; Sordoni, A.; and Agarwal, R. 2024. V-star: Training verifiers for self-taught reasoners. *arXiv preprint arXiv:2402.06457*.

Kong, A.; Ma, W.; Zhao, S.; Li, Y.; Wu, Y.; Wang, K.; Liu, X.; Li, Q.; Qin, Y.; and Huang, F. 2025. SDPO: Segment-Level Direct Preference Optimization for Social Agents. *arXiv preprint arXiv:2501.01821*.

Kostric, I.; Balog, K.; and Radlinski, F. 2024. Generating usage-related questions for preference elicitation in conversational recommender systems. *ACM Transactions on Recommender Systems*, 2(2): 1–24.

Li, B. Z.; Tamkin, A.; Goodman, N.; and Andreas, J. 2025. Eliciting Human Preferences with Language Models. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Lin, J.; Tomlin, N.; Andreas, J.; and Eisner, J. 2024. Decision-oriented dialogue for human-ai collaboration. *Transactions of the Association for Computational Linguistics*, 12: 892–911.

Liu, Q.; Wu, X.; Zhao, X.; Zhu, Y.; Xu, D.; Tian, F.; and Zheng, Y. 2024. When moe meets llms: Parameter efficient fine-tuning for multi-task medical applications. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1104–1114.

Loepp, B.; and Ziegler, J. 2024. Exploring the potential of generative ai for augmenting choice-based preference elicitation in recommender systems. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*, 114–119.

Madotto, A.; Lin, Z.; Winata, G. I.; and Fung, P. 2021. Few-shot bot: Prompt-based learning for dialogue systems. *arXiv preprint arXiv:2110.08118*.

Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1: 3.

Mayer, C. W.; Ludwig, S.; and Brandt, S. 2023. Prompt text classifications with transformer models! An exemplary introduction to prompt-based learning with large language models. *Journal of Research on Technology in Education*, 55(1): 125–141.

Occhipinti, D.; Tekiroglu, S. S.; and Guerini, M. 2023. Prodigy: a profile-based dialogue generation dataset. *arXiv preprint arXiv:2311.05195*.

Park, C.; Donahue, K.; and Raghavan, M. 2025. When to Ask a Question: Understanding Communication Strategies in Generative AI Tools. In *Adjunct Proceedings of the 33rd ACM Conference on User Modeling, Adaptation and Personalization*, 288–299.

Piriyakulkij, W. T.; Kuleshov, V.; and Ellis, K. 2023. Active preference inference using language models and probabilistic reasoning. *arXiv preprint arXiv:2312.12009*.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36: 53728–53741.

Reynolds, L.; and McDonell, K. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended abstracts of the 2021 CHI conference on human factors in computing systems*, 1–7.

Shi, W.; Yuan, M.; Wu, J.; Wang, Q.; and Feng, F. 2024. Direct multi-turn preference optimization for language agents. *arXiv preprint arXiv:2406.14868*.

Song, Y.; Yin, D.; Yue, X.; Huang, J.; Li, S.; and Lin, B. Y. 2024. Trial and error: Exploration-based trajectory optimization for llm agents. *arXiv preprint arXiv:2403.02502*.

Tamkin, A.; Handa, K.; Shrestha, A.; and Goodman, N. 2023. Task Ambiguity in Humans and Language Models. In *The Eleventh International Conference on Learning Representations (ICLR)*.

Wang, J.; Shi, E.; Yu, S.; Wu, Z.; Ma, C.; Dai, H.; Yang, Q.; Kang, Y.; Wu, J.; Hu, H.; et al. 2023. Prompt engineering for healthcare: Methodologies and applications. *arXiv preprint arXiv:2304.14670*.

Xing, M.; Zhang, R.; Xue, H.; Chen, Q.; Yang, F.; and Xiao, Z. 2024. Understanding the weakness of large language model agents within a complex android environment. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 6061–6072.

Yu, X.; Zhou, C.; Fang, Y.; and Zhang, X. 2024. Multigprompt for multi-task pre-training and prompting on graphs. In *Proceedings of the ACM Web Conference 2024*, 515–526.

Zelikman, E.; Harik, G.; Shao, Y.; Jayasiri, V.; Haber, N.; and Goodman, N. D. 2024. Quiet-star: Language models can teach themselves to think before speaking. *arXiv preprint arXiv:2403.09629*.

Zelikman, E.; Wu, Y.; Mu, J.; and Goodman, N. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35: 15476–15488.

Zhang, Z.; Yao, Y.; Zhang, A.; Tang, X.; Ma, X.; He, Z.; Wang, Y.; Gerstein, M.; Wang, R.; Liu, G.; et al. 2025. Igniting language intelligence: The hitchhiker's guide from chain-of-thought reasoning to language agents. *ACM Computing Surveys*, 57(8): 1–39.