

LQ-Rewriter: Generative Reward Model-Guided Lazy Query Rewriting for Role-Playing Dialogue

Wei Song¹, Zheng Huang¹, Renyu Zhu^{3*}, Runze Wu³, Zhenya Huang^{1,2†},
Yihuan Mao³, Rongsheng Zhang³, Tangjie Lv³, Kai Zhang^{1,2}

¹State Key Laboratory of Cognitive Intelligence, University of Science and Technology of China

²Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

³NetEase Fuxi AI Lab

{sw2, huangzheng}@mail.ustc.edu.cn, {huangzhy, kkzhang08}@ustc.edu.cn,

{zhurenryu, wurunze1, maoyihuan01, zhangrongsheng, hzlv tangjie}@corp.netease.com

Abstract

With the rapid development of large language models (LLMs), role-playing dialogue systems have become increasingly popular for immersive and personalized human-AI interaction. However, in real-world role-playing scenarios, users often provide *lazy queries*—brief and ambiguous responses such as “oh,” “hmm,” or “?”—which lack explicit intent and emotional cues, making it difficult for role-playing LLMs to generate engaging replies. Existing query rewriting methods mainly target functional tasks such as information retrieval and are not tailored to role-playing dialogue, where rewrite quality depends on role-playing-specific factors such as character consistency and personalized user preferences. To address this, we first propose a listwise Generative Reward Model (GRM) trained from real user preference data collected through online regeneration behavior, which provides fine-grained process and final preference signals for training a **Lazy Query Rewriter**. Using rejection fine-tuning and reinforcement learning, the LQ-Rewriter learns to transform underspecified user inputs into richer expressions while preserving user intent and dialogue context. Experiments on real-world dialogue logs from a mainstream Chinese AI-powered romantic role-playing platform show that our GRM substantially outperforms general-purpose LLM judges. Moreover, our LQ-Rewriter achieves the highest GRM-based and human preference win rates among all rewriting baselines, and brings tangible improvements in online deployment.

1 Introduction

With the maturation of large language models (LLMs), role-playing dialogue (Shao et al., 2023; Wang et al., 2024a, 2025b; Du et al., 2026) has emerged as an important paradigm for human-AI interaction. Whether in virtual companionship,

*Corresponding author

†Corresponding author

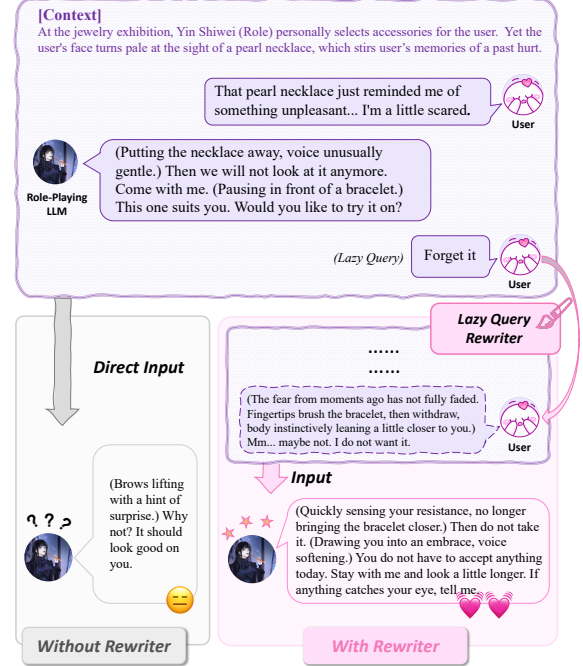


Figure 1: From Lazy Query to Better Role-playing. **Without LQ-Rewriter:** (1) Weak Intent Signal (2) Flat Interaction (3) Conversation Stalls. **With LQ-Rewriter:** (1) Intent Amplified (2) Richer Emotion (3) Longer Conversation. Chinese version can be seen in Figure 4.

narrative role-play, or casual entertainment, users increasingly expect immersive and personalized conversational experiences shaped by LLM-driven characters (Chen et al., 2023; Yuan et al., 2024; Dai et al., 2025; Wang et al., 2026). Unlike functional dialogue systems (e.g., information retrieval or task-oriented interactions) that heavily prioritize factual correctness, the core of role-playing dialogue lies in emotional engagement and sustained interaction. It relies on natural and coherent multi-turn conversations to maintain user interest. This, in turn, requires models to dynamically adapt their response strategies based on context—for example, advancing the conversation while preserving character consistency.

Nevertheless, in real-world interactions, role-playing scenarios suffer from a critical pain point: users’ core demand is a casual, effortless experience rather than precise, elaborate expression. Driven by (1) a desire to avoid cognitively costly responses, (2) a lack of clear expressive intent, or (3) a preference for passive engagement, users frequently produce extremely brief and ambiguous utterances such as “oh”, “hmm”, “?”, or simply select prefabricated responses provided by the platform. These *lazy queries* typically lack explicit semantic guidance and affective cues, making it challenging for LLMs to infer users’ true interactive intentions. In practice, we observe that such ambiguity often leads to low-quality or context-misaligned responses, which in turn weakens dialogue continuity and undermines the immersive experience of role-playing.

To address this issue, lazy-query rewriting serves as a promising direction: by enriching underspecified user inputs, a rewriter can provide clearer interaction signals for the downstream role-playing model. However, existing query rewriting methods are not well-suited for role-playing scenarios. Approaches such as HyDE (Gao et al., 2023) and ConvSearch-R1 (Zhu et al., 2025) are designed for information retrieval and other functional tasks, focusing on semantic matching or answer correctness rather than role-playing-specific factors such as character consistency, emotional engagement, narrative progression, and personalized user preferences. A further challenge lies in how to train such a role-playing-oriented rewriter. Unlike functional rewriting tasks, where rewrite quality can often be measured by retrieval accuracy or answer correctness, the quality of a lazy-query rewrite should be judged by whether it improves the downstream role-playing response and better matches user preferences. This requires reliable preference-aligned reward signals. However, evaluation in role-playing dialogue remains underexplored due to its reliance on subjective user experience. Current approaches often rely on LLM-as-a-judge to evaluate individual responses, which suffers from limitations such as neglecting personalized preferences and insufficient adaptability of evaluation strategies. In addition, most methods adopt pointwise scoring, which may introduce bias in reward estimation.

To overcome these limitations, we first propose a listwise Generative Reward Model (GRM) (Liu et al., 2025b; Jia et al., 2025), trained on real user preference data collected via an online regeneration

mechanism, and further optimized through rejection fine-tuning (RFT) and reinforcement learning (RL). Given the same dialogue context, the GRM compares multiple candidate responses generated by role-playing models and produces fine-grained process and personalized preference scores. Rather than directly judging rewritten queries in isolation, we use the GRM to evaluate the downstream role-playing responses elicited by different rewrites, thereby providing preference-aligned supervision and reward signals for training the rewriter. With this reward signal, we further develop a **Lazy-Query Rewriter** tailored to role-playing scenarios. Specifically, the LQ-Rewriter follows a “*plot progression + micro-level writing style*” strategy to transform *lazy queries* into richer and more context-aware expressions. LQ-Rewriter is also trained with RFT and RL. During RFT, a teacher model generates multiple candidate rewrites, and the GRM selects the rewrite whose downstream response best matches user preferences. During RL, the GRM provides comparative rewards by comparing responses generated from rewritten queries against those generated from the original lazy query. Through this two-stage training process, the LQ-Rewriter learns to produce rewrites that better preserve user intent, fit the dialogue context, and improve the quality of downstream role-playing responses. In addition, we develop a comprehensive multi-dimensional evaluation framework for the LQ-Rewriter, covering expert-designed criteria, GRM-based preference alignment and online interaction metrics. Experimental results show that our approach outperforms strong baselines and brings tangible improvements in real-world deployment. Our main contributions are summarized as follows:

1. We propose a listwise GRM trained on real user preference data collected through an online regeneration mechanism. It captures fine-grained preferences in role-playing scenarios and outperforms general-purpose LLMs.
2. We introduce LQ-Rewriter, a role-playing-oriented *lazy-query* rewriter guided by GRM. It follows a “*plot progression + micro-level writing style*” strategy, enabling rewrites that improve downstream role-playing responses.
3. We conduct comprehensive multi-dimensional evaluations of the proposed LQ-Rewriter, demonstrating its superior performance in role-playing scenarios.

2 Related Work

2.1 Reward Modeling for Role-Playing

Recent advances in large language models have enabled increasingly immersive and personalized role-playing agents (Tseng et al., 2024; Chen et al., 2024a; Zhang et al., 2026; Xu et al., 2026; Tang et al., 2026a), attracting growing attention from both academia and industry (Chen et al., 2024b; Liu et al., 2024; Qin et al., 2025a; Lu et al., 2026; Tang et al., 2026b). A central challenge in this setting is how to evaluate role-playing responses, whose quality depends not only on role-playing-specific factors such as character consistency and dialogue coherence, but also on subjective user experience. Existing role-playing evaluation methods commonly rely on LLM-as-a-judge to score individual responses (Wang et al., 2024b; Feng et al., 2025; Zhou et al., 2025; He et al., 2025; Lu et al., 2025). For example, CharacterEval (Tu et al., 2024) evaluates responses across predefined dimensions such as conversational ability, character consistency, role-playing engagement, and persona backtracking, while RMTBench (Xiang et al., 2025) introduces user-centric criteria including intent satisfaction, dialogue coherence, and character appropriateness. Although these methods cover important aspects of role-playing quality, they still rely on relatively static evaluation dimensions and may fail to adapt to diverse dialogue contexts or personalized user preferences. Moreover, most of them adopt pointwise scoring, where each response is evaluated independently, which can lead to unstable absolute scores and biased reward estimation. Another line of work explores comparative reward modeling through pairwise or listwise evaluation. ChARM (Fang et al., 2025) derives pairwise preference signals through behavior-adaptive optimization, while CPO (Ye et al., 2025) reformulates reward evaluation as group-level comparison. Nevertheless, these methods still largely depend on static evaluation dimensions and general LLM judges to produce reward signals.

2.2 Query Rewriting

Query rewriting has been widely studied to transform user inputs into more informative or better-structured queries for downstream systems. For example, HyDE (Gao et al., 2023) improves dense retrieval by generating hypothetical documents, while ConvSearch-R1 (Zhu et al., 2025) and TongSearch QR (Qin et al., 2025b) optimize

query rewriting with retrieval-oriented reward signals. Beyond retrieval, Align-Pro (Trivedi et al., 2025) learns optimized prompts for LLM alignment, and PromptEnhancer (Wang et al., 2025a) rewrites prompts to improve image-text alignment in text-to-image generation. Despite their effectiveness, these methods are designed for functional tasks such as information retrieval, question answering or text-to-image generation, and not well-suited for role-playing scenarios.

2.3 Reinforcement Learning for LLM Fine-Tuning

Reinforcement learning is widely used for fine-tuning LLMs to better align with user preferences. Methods such as PPO (Schulman et al., 2017) and its variants (e.g., DPO (Rafailov et al., 2023), GRPO (Shao et al., 2024), DAPO (Yu et al., 2026)) optimize model policies based on reward signals. In role-playing scenarios, approaches like CPO (Ye et al., 2025) further extend this paradigm. In this work, we apply GRPO (see Appendix A.1) to train the GRM, and adopt CPO (see Appendix A.2) to optimize the LQ-Rewriter.

3 Methodology

3.1 Overview

First, we collect real-world user interaction data through an online regeneration mechanism¹ to construct a listwise preference dataset for role-playing scenarios. Based on this dataset, we train a generative reward model (GRM) on top of a base model via rejection fine-tuning (RFT) and reinforcement learning (RL), enabling it to capture fine-grained user preferences and assign scores to multiple candidate responses generated under the same context.

Furthermore, we adopt a two-stage approach to train the LQ-Rewriter. In the RFT stage, based on real dialogue data, we employ a teacher model to generate diverse rewriting candidates using a customized prompt that combines plot progression and micro-level rewriting style, and construct high-quality supervision data by selecting the best candidates with the GRM. In the RL stage, we further optimize the rewriter using a CPO-based (Ye et al.,

¹We emphasize the importance of learning from real online user behavior rather than relying solely on manual annotations, especially in romantic role-playing scenarios. User preferences in this domain are highly diverse and subjective: the same response may be preferred by some users but disliked by others. As a result, human-labeled data may primarily reflect the preferences of a small group of annotators and may not generalize well to real online users.

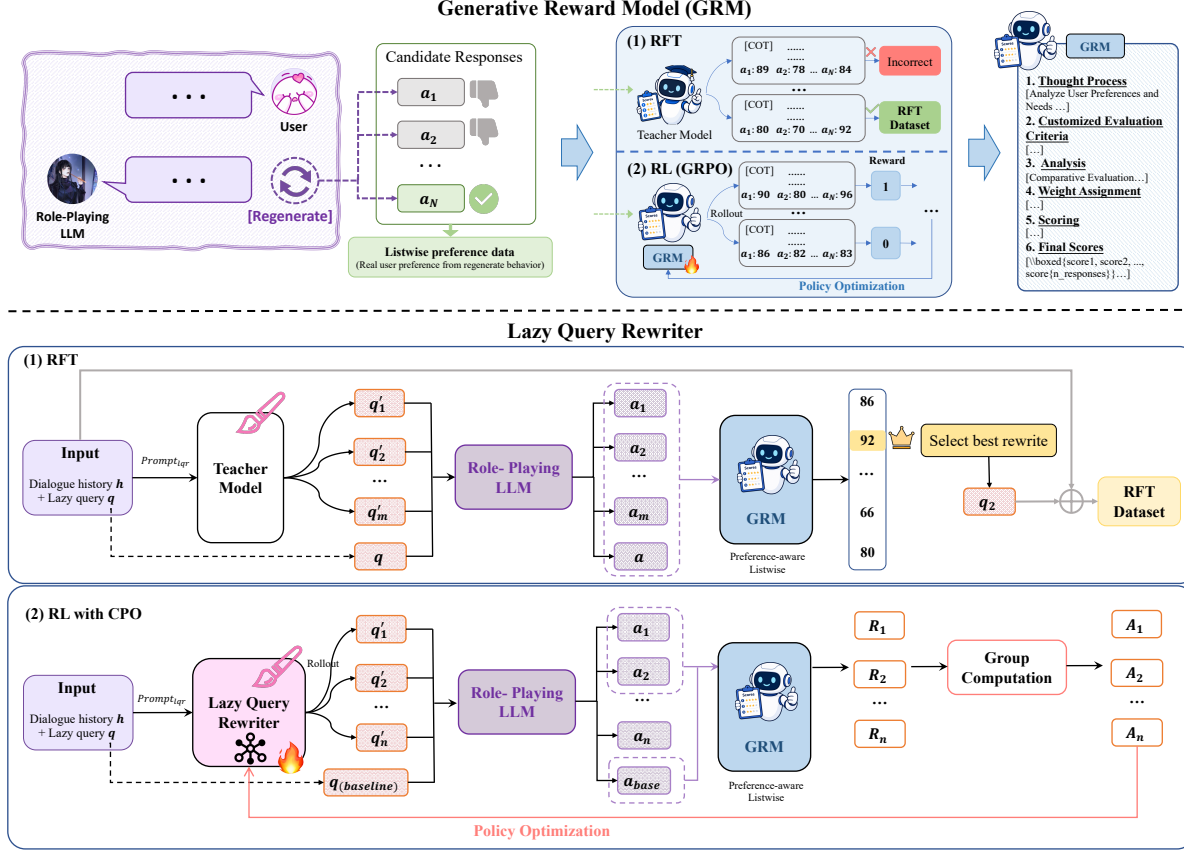


Figure 2: Framework overview. We first train a listwise GRM from online user preferences collected through the regeneration mechanism, enabling fine-grained preference scoring for role-playing responses. The GRM then guides LQ-Rewriter training by selecting teacher-generated rewrites for RFT and providing baseline-comparative rewards for CPO. The final LQ-Rewriter enriches underspecified user inputs to improve downstream role-playing responses.

2025) strategy, where the original query serves as a baseline and the GRM provides relative reward signals for policy optimization, guiding the model toward generating better rewrites. During inference, the LQ-Rewriter transforms the user’s original lazy query into a more informative input, which is then fed into the role-playing model to generate responses, thereby improving conversational immersion and overall interaction quality.

3.2 Generative Reward Model (GRM)

3.2.1 Data Construction

We construct a listwise preference dataset for role-playing scenarios using an online regeneration mechanism, which captures implicit user preferences during real interactions.

Concretely, given a dialogue context x , the role-playing model generates a sequence of candidate responses through iterative regeneration triggered by user feedback. Let $\mathcal{A} = \{a_1, a_2, \dots, a_N\}$, $2 \leq N \leq 5$ denote the set of candidate responses obtained within a single interaction, where each a_i

corresponds to a response generated after the i -th regeneration. When the user stops regenerating, the final response a_N is treated as the implicitly preferred (positive) sample, while the remaining responses are treated as negative samples.

To prevent the model from exploiting positional bias, we randomly shuffle the order of candidates in $\mathcal{A}$, while preserving the identity of the unique positive sample. Each training instance is then represented as: $(x, \mathcal{A}, y^*)$, where $y^* \in \mathcal{A}$ denotes the ground-truth preferred response.

We further render each instance into a structured prompt using a task-specific template $\text{PROMPT}_{\text{grm}}$ (see Appendix E.1) designed for role-playing evaluation, which serves as the input to the GRM. The resulting dataset is split into three subsets: $\mathcal{D}_{\text{rft}}^{\text{grm}}$, $\mathcal{D}_{\text{rl}}^{\text{grm}}$, $\mathcal{D}_{\text{test}}^{\text{grm}}$.

3.2.2 Rejection Fine-Tuning

To enable effective supervision, we first construct high-quality training targets via rejection sampling, followed by rejection fine-tuning (RFT).

Rejection Sampling Since the raw dataset does not contain explicit model outputs for supervision, we employ a teacher model f_{teacher} to generate candidate annotations. For each input instance $(x, \mathcal{A}, y^*)$, we query the teacher model multiple times to obtain a set of outputs: $\{o^{(1)}, o^{(2)}, \dots, o^{(K)}\}$.

Each output $o^{(k)}$ contains a score assignment over the candidate set $\mathcal{A}$. We define a binary reward function:

$$r(o^{(k)}) = \begin{cases} 1, & \text{if } \arg \max_{a \in \mathcal{A}} s^{(k)}(a) = y^* \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $s^{(k)}(a)$ denotes the score assigned to candidate a in output $o^{(k)}$.

If there exists at least one $o^{(k)}$ such that $r(o^{(k)}) = 1$, we randomly select one such output as the supervision target. Otherwise, we optionally augment the prompt with a hint indicating the ground-truth preferred response and re-query the teacher model. Only samples with valid (correct) annotations are retained, resulting in a filtered dataset: $\tilde{\mathcal{D}}_{\text{rft}}^{\text{grm}}$.

RFT Objective We then fine-tune the GRM using standard cross-entropy loss:

$$\mathcal{L}_{\text{RFT}} = -\mathbb{E}_{(x, \mathcal{A}, o) \sim \tilde{\mathcal{D}}_{\text{rft}}^{\text{grm}}} \log p_{\theta}(o \mid x, \mathcal{A}), \quad (2)$$

where p_{θ} denotes the GRM parameterized by θ , and o is the selected supervision output.

3.3 Reinforcement Learning

To further enhance the GRM’s ability to model user preferences, we initialize the GRM policy from the RFT-trained checkpoint and further optimize it using Group Relative Policy Optimization (GRPO) on $\mathcal{D}_{\text{rl}}^{\text{grm}}$.

Given an input $(x, \mathcal{A}, y^*)$, the GRM samples multiple rollout outputs:

$$\{o^{(1)}, o^{(2)}, \dots, o^{(R)}\} \sim \pi_{\theta}(\cdot \mid x, \mathcal{A}), \quad (3)$$

where each rollout output $o^{(r)}$ contains the generated evaluation process and score assignments over the candidate responses.

For each rollout, we parse the predicted preference ranking and define the reward function as:

$$r(o^{(r)}) = \begin{cases} 1, & \text{if } \arg \max_{a \in \mathcal{A}} s^{(r)}(a) = y^* \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $s^{(r)}(a)$ denotes the score assigned to candidate response a in rollout $o^{(r)}$.

Following the GRPO framework, we optimize the GRM policy using a PPO-style clipped policy objective with group-relative advantages and KL regularization. This optimization encourages the model to assign higher scores to the ground-truth preferred response within each candidate set, thereby improving its ability to capture fine-grained user preferences in a listwise manner.

3.4 LQ-Rewriter Training

The LQ-Rewriter is designed to handle *lazy queries* by transforming underspecified user inputs into more informative and contextually grounded expressions. Specifically, given a dialogue history h and a user lazy query q , the LQ-Rewriter generates an enriched query q' , which is then combined with h and fed into the role-playing model to produce a response. This process improves conversational immersion and interaction quality. For practical deployment, we adopt a relatively small base model as the backbone of the LQ-Rewriter to ensure efficiency.

3.4.1 Data Construction

We construct the LQ-Rewriter training data from real-world dialogue logs containing user *lazy queries*. Each instance is defined as a pair: (h, q) , where h denotes the dialogue history and q is the user’s lazy query.

We design a task-specific rewriting prompt template, denoted as $\text{PROMPT}_{\text{lqr}}$ (see Appendix E.2), which guides the model to generate rewritten queries following a combined strategy of “*plot progression + micro-level writing style*”. Each pair (h, q) is rendered into a model input using this template. The dataset is then split into: $\mathcal{D}_{\text{rft}}^{\text{lqr}}, \mathcal{D}_{\text{rl}}^{\text{lqr}}, \mathcal{D}_{\text{test}}^{\text{lqr}}$.

3.4.2 Rejection Fine-Tuning

Since ground-truth rewritten queries are unavailable, we construct supervision signals via a generate-and-select paradigm guided by the pre-trained GRM.

Multi-Candidate Query Generation For each input (h, q) , we use a teacher model to generate a set of diverse rewritten queries: $\mathcal{Q} = \{q'_1, q'_2, \dots, q'_m\}$.

Preference-Based Selection Each rewritten query q'_i is combined with the dialogue history h

and fed into the role-playing model $\mathcal{M}$ to generate a response:

$$a_i = \mathcal{M}(h, q'_i). \quad (5)$$

Let $\mathcal{A} = \{a_1, \dots, a_m\}$ denote the set of candidate responses. We evaluate them using the GRM in a listwise manner:

$$\{s_i\}_{i=1}^m = \text{GRM}((h, q), \mathcal{A}). \quad (6)$$

The rewritten query corresponding to the highest-scoring response is selected as the optimal rewrite:

$$q^* = \arg \max_{q'_i \in \mathcal{Q}} s_i. \quad (7)$$

This yields a high-quality training dataset: $\tilde{\mathcal{D}}_{\text{rft}}^{\text{gr}} = \{(h, q, q^*)\}$, which is used to fine-tune the LQ-Rewriter.

3.4.3 Reinforcement Learning for LQ-Rewriter

We initialize the rewriter policy from the RFT-trained checkpoint and further optimize it using Comparative Policy Optimization (CPO) to improve alignment with user preferences. Unlike GRPO, which assigns rewards independently to each candidate, CPO adopts a comparative evaluation paradigm, where rewards are computed based on relative preferences within a group of candidates.

Given an input (h, q) , we sample n rewritten queries via rollout: $\{q'_1, q'_2, \dots, q'_n\}$. These candidates are partitioned into batches of size up to G_b , where each batch, together with the original query q , forms a comparison group. For each query in the group, we generate responses using the role-playing model:

$$a_i = \mathcal{M}(h, q'_i), \quad (8)$$

$$a_{\text{base}} = \mathcal{M}(h, q). \quad (9)$$

All responses within the group are jointly evaluated by the GRM to obtain preference scores. We define a relative reward function based on comparison with the baseline:

$$r(q'_i) = \begin{cases} 1, & \text{if } s_i > s_{\text{base}} \\ 0, & \text{if } s_i = s_{\text{base}} \\ -1, & \text{if } s_i < s_{\text{base}} \end{cases}. \quad (10)$$

The rewriter policy is then optimized using these relative rewards, encouraging it to generate rewritten queries that consistently lead to better responses than the original lazy query.

4 Experimental Setup

4.1 Datasets

We construct different datasets for training the GRM and the LQ-Rewriter, respectively. All datasets are constructed from real-world role-playing dialogue logs collected from the online platform for romantic role-playing. Table 1 summarizes the dataset statistics. For GRM training, since multiple role-playing models are deployed online, the collected data involve responses generated by different role-playing models, including Doubao-1.5-pro-32k-character-250715, Qwen-Plus-Character, and others. More model details are provided in Appendix B.

Table 1: Statistics of datasets used for GRM and LQ-Rewriter training.

Dataset	RFT	RL	Test
GRM	10,000	10,000	910
LQ-Rewriter	18,286	10,000	1,500

4.2 Models

GRM We adopt Qwen3-32B² (Yang et al., 2025) as the base model and Qwen3-235B-A22B-Instruct-2507³ as the teacher model.

LQ-Rewriter We use Qwen3-4B⁴ and Qwen3-14B⁵ as base models, with DeepSeek-V3.2-Think⁶ (Liu et al., 2025a) as the teacher model. The role-playing responses are generated using Qwen-Plus-Character.

4.3 Baselines

GRM baselines: (1) Random Guess (2) GPT-5 (3) DeepSeek-R1 (4) Doubao-Seed-1.6-Thinking (5) Gemini3-Pro-Preview.

LQ-Rewriter baselines: (1) **Original w/o Rewrite**, which directly feeds the original lazy query into the role-playing model; (2) **large-scale general-purpose LLM**, including GPT-5.5, Gemini-3-Pro, Qwen3.6-Max-Preview, DeepSeek-V4-Pro, and Doubao-Seed-2.0-Pro; (3) **lightweight general-purpose LLM**, including GPT-5.4-Mini, Gemini-3.5-Flash, and Qwen3.6-Flash; (4) **teacher**

²<https://huggingface.co/Qwen/Qwen3-32B>

³<https://huggingface.co/Qwen/Qwen3-235B-A22B-Instruct-2507>

⁴<https://huggingface.co/Qwen/Qwen3-4B>

⁵<https://huggingface.co/Qwen/Qwen3-14B>

⁶<https://huggingface.co/deepseek-ai/DeepSeek-V3.2>

model, DeepSeek-V3.2-Think, which is used to generate rewriting candidates during data construction; (5) **base model**, Qwen3-14B; (6) **RFT-only variant**, Qwen3-14B+RFT, which is trained with rejection fine-tuning but without RL optimization.

The source links of these models are provided in Appendix C.

4.4 Evaluation Metrics

4.4.1 GRM Evaluation

We evaluate GRM using accuracy on $\mathcal{D}_{\text{test}}^{\text{grm}}$, defined as whether the model correctly assigns the highest score to the ground-truth preferred response.

4.4.2 LQ-Rewriter Evaluation

We evaluate the LQ-Rewriter from multiple perspectives:

Non-Rewriting Rate (NR) Measures the proportion of outputs where the model directly answers the query instead of rewriting it.

Groundedness (Ground.) Evaluates whether the rewritten query preserves the original user intent.

Contextual Relevance (CtxRel.) Contextual relevance measures how well the rewritten query aligns with the role-playing context, including the character setting, dialogue history, emotional state, and narrative flow.

Diversity (Div.) We evaluate rewriting diversity using two automatic metrics: Distinct-2 and Self-BLEU. Distinct-2 measures the proportion of unique bigrams in generated rewrites, and Self-BLEU measures similarity among generated rewrites.

GRM Win Rate (GRM-W) We compare rewritten queries with the original query by feeding their resulting responses into the GRM. A *win* is counted when the rewritten query leads to a response preferred over the original-query response.

Human Preference Win Rate (Human-W) We conduct pairwise comparisons using human evaluators (= 3 experts familiar with romantic role-playing scenarios). They evaluate rewritten vs. original responses based on: (i) subjective experience (ii) contextual appropriateness. Human-W is the proportion of majority-vote *wins*.

Online Metrics We evaluate real-world impact using: Average daily conversation turns per user (7-day window). In our online platform, conversation

turns are a key engagement metric, as they are closely tied to the platform’s business objective.

4.5 Implementation Details

All models are trained under the VeRL framework on 8 NVIDIA H200 GPUs. For GRM training, we use 8 samples for both RFT rejection sampling and RL rollout. For the LQ-Rewriter, we use 4 samples for RFT and 8 rollouts for RL, with a batch size of 4 for CPO grouping (see Appendix D for details).

Table 2: GRM Preference Prediction Accuracy.

Model	Accuracy	Δ Random
Random Guess	32.32%	–
GPT-5	27.25%	↓ 5.07%
DeepSeek-R1	32.23%	↓ 0.09%
Doubao-Seed-1.6-Thinking	29.78%	↓ 2.54%
Qwen3-235B-A22B-Instruct	29.04%	↓ 3.28%
Gemini3-Pro-Preview	36.02%	↑ 3.70%
GRM-32B (Ours)	73.85%	↑ 41.53%

5 Experimental Results

5.1 GRM Performance

Table 2 shows the preference prediction accuracy of different models on $\mathcal{D}_{\text{test}}^{\text{grm}}$. All models are evaluated under the same input setting. Our GRM-32B achieves the best accuracy of 73.85%, outperforming the random-guess baseline⁷ by 41.53 % and significantly exceeding all baselines. In contrast, most strong general-purpose LLMs perform close to or even below this random-guess baseline, suggesting that they struggle to capture fine-grained and highly subjective user preferences in romantic role-playing dialogues. These results further demonstrate the effectiveness of our GRM in modeling personalized user preferences in this scenario.

5.2 LQ-Rewriter Performance

Table 3 presents the evaluation results of different rewriting methods on $\mathcal{D}_{\text{test}}^{\text{lqr}}$. Overall, our LQ-Rewriter variants achieve the strongest preference-based performance against the original w/o rewrite baseline. Specifically, LQ-Rewriter-4B and LQ-Rewriter-14B obtain GRM win rates of 60.7% and

⁷For the random-guess baseline, the expected accuracy depends on the number of candidate responses in each sample. Specifically, the test set contains 26.26% samples with 2 candidates, 23.30% with 3 candidates, 26.15% with 4 candidates, and 24.29% with 5 candidates. Therefore, the expected random-guess accuracy is computed as: $0.2626 \times \frac{1}{2} + 0.2330 \times \frac{1}{3} + 0.2615 \times \frac{1}{4} + 0.2429 \times \frac{1}{5} = 0.3232$, which corresponds to an overall random-guess accuracy of 32.32%.

Table 3: Performance comparison on $\mathcal{D}_{\text{test}}^{\text{lqr}}$. $\downarrow$ indicates the lower, the better. $\uparrow$ indicates the higher, the better.

Model	NR $\downarrow$	Ground. $\uparrow$	CtxRel. $\uparrow$	Div.		GRM-W $\uparrow$	Human-W $\uparrow$
				Dist-2 $\uparrow$	Self-BLEU $\downarrow$		
Original w/o Rewrite	–	–	–	–	–	baseline	baseline
GPT-5.5	1.9%	4.74	4.90	0.373	58.9	53.3%	54.9%
Gemini-3-Pro	0.1%	4.60	4.75	0.374	51.0	54.0%	55.9%
Qwen3.6-Max-Preview	0.7%	4.14	4.40	0.306	76.6	51.7%	51.1%
DeepSeek-V4-Pro	3.0%	4.51	4.72	0.569	20.9	49.9%	50.8%
Doubao-Seed-2.0-Pro	0.1%	4.87	4.92	0.449	46.7	54.9%	55.2%
GPT-5.4-Mini	2.3%	4.48	4.67	0.426	46.8	51.0%	51.9%
Gemini-3.5-Flash	0.0%	4.62	4.77	0.423	39.9	53.7%	54.2%
Qwen3.6-Flash	0.2%	4.34	4.56	<u>0.534</u>	<u>21.2</u>	52.6%	54.4%
DeepSeek-V3.2-Think	6.5%	4.81	4.88	0.442	43.9	51.0%	52.7%
Qwen3-14B	0.0%	3.73	4.09	0.425	45.0	48.8%	49.0%
Qwen3-14B+RFT	2.3%	4.39	4.67	0.498	31.8	53.6%	54.4%
LQ-Rewriter-4B	0.0%	4.52	4.56	0.199	67.5	60.7%	60.6%
LQ-Rewriter-14B	0.0%	4.71	4.82	0.249	70.7	<u>59.8%</u>	<u>60.1%</u>

59.8%, respectively, outperforming all other methods. A similar trend is observed in human evaluation, where LQ-Rewriter-4B and LQ-Rewriter-14B achieve Human-W scores of 60.6% and 60.1%, respectively. These results demonstrate that GRM-guided RL can effectively improve the rewriter and produce rewrites that lead to better downstream role-play responses. Both LQ-Rewriter variants also achieve a 0.0% non-rewriting rate, indicating strong compliance with the rewriting instruction. In terms of groundedness and contextual relevance, LQ-Rewriter-14B performs better than LQ-Rewriter-4B, suggesting that larger model capacity helps preserve user intent and contextual coherence. Meanwhile, we observe that our LQ-Rewriters show lower Distinct-2 and higher Self-BLEU than some API baselines, indicating a certain degree of stylistic concentration after preference optimization. Nevertheless, the leading Human-W scores show that this reduced diversity does not hurt human preference in our evaluation.

Online Results We further evaluate the LQ-Rewriter in an online setting. As shown in Figure 3, incorporating the rewriter leads to an increase in the average number of daily conversation turns per user, demonstrating improved user engagement in real-world interactions.

6 Conclusion

In this paper, we study the problem of *lazy-query* rewriting in role-playing dialogue, where users often provide brief and ambiguous inputs which may

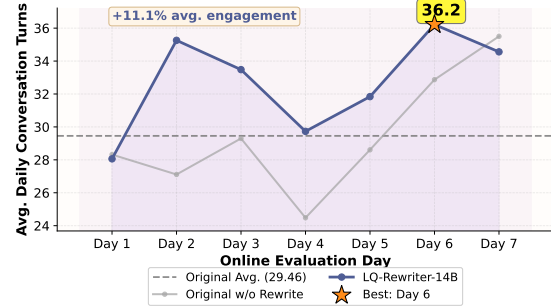


Figure 3: Online evaluation over a 7-day window. The star marks the best daily performance of LQ-Rewriter-14B, and the dashed line denotes the average performance of the original system without rewriting.

reduce the quality of role-playing responses. To address this challenge, we propose a listwise Generative Reward Model (GRM) trained from real user preference data collected through an online regeneration mechanism. The GRM models fine-grained user preferences over multiple candidate responses and provides preference-aligned supervision and reward signals for training the rewriter. Based on the GRM, we further introduce LQ-Rewriter, a role-playing-oriented *lazy-query* rewriter that follows a “*plot progression + micro-level writing style*” strategy and is trained through rejection fine-tuning and reinforcement learning. Extensive experiments demonstrate that our GRM substantially outperforms general-purpose LLM judges in preference prediction, and that LQ-Rewriter achieves strong performance across multiple metrics and brings tangible improvements in online deployment.

Limitations

First, this work focuses on improving role-playing interactions from the input side through lazy-query rewriting, rather than directly enhancing the underlying role-playing models themselves. Since the role-playing models deployed in our online platform are API-based models, we treat them as downstream response generators and investigate how to provide them with more informative user inputs. Future work could incorporate the insights from our framework into the training or alignment of role-playing models themselves, so that they can better understand and respond to these *lazy queries*. Second, the regeneration-based preference labels used to train the GRM are implicit behavioral signals and may contain noise. Although stopping regeneration does not always indicate a globally optimal response, this mechanism enables large-scale preference data collection from natural user interactions without disrupting the user experience. Future work could explore explicit candidate-selection mechanisms to further validate and calibrate these implicit preference signals. Third, the reward signal used in our framework mainly comes from the trained GRM, which still leaves room for further improvement. A promising direction is to develop composite reward signals that combine GRM scores with additional online behavioral feedback. For example, the number of subsequent dialogue turns after a response could be incorporated as complementary rewards.

Ethical Considerations

The data used in this work are collected from an online role-playing platform under the platform’s user agreement and privacy policy, where users explicitly consent to the use of anonymized interaction data for platform improvement and research purposes. To protect user privacy, all personally identifiable information is removed during data processing, and the study only uses anonymized conversational data for research purposes. We do not disclose any user identities or private information in this paper. Due to privacy and safety considerations, the collected dataset will not be publicly released. However, we will release the training code at <https://github.com/Mercidaiha/LQ-Rewriter> to support future research.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grants No.62477044), the Key Technologies R&D Program of Anhui Province (No.202423k09020039), the Young Elite Scientists Sponsorship Program by CAST (No.2024QNRC001), the Fundamental Research Funds for the Central Universities (No.WK2150110038), and CCF-NetEase ThunderFire Innovation Research Funding (NO.CCF-Netease 202512).

References

- Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, et al. 2024a. From persona to personalization: A survey on role-playing language agents. *arXiv preprint arXiv:2404.18231*.
- Nuo Chen, Yan Wang, Yang Deng, and Jia Li. 2024b. The oscars of ai theater: A survey on role-playing with language models. *arXiv preprint arXiv:2407.11484*.
- Nuo Chen, Yan Wang, Haiyun Jiang, Deng Cai, Yuhao Li, Ziyang Chen, Longyue Wang, and Jia Li. 2023. Large language models meet harry potter: A dataset for aligning dialogue agents with characters. In *Findings of the association for computational linguistics: EMNLP 2023*, pages 8506–8520.
- Yanqi Dai, Huanran Hu, Lei Wang, Shengjie Jin, Xu Chen, and Zhiwu Lu. 2025. Mmrole: A comprehensive framework for developing and evaluating multimodal role-playing agents. In *International Conference on Learning Representations*, volume 2025, pages 66300–66321.
- Chengyu Du, Xintao Wang, Aili Chen, Weiyan Li, Rui Xu, Junteng Liu, Zishan Huang, Rong Tian, Zijun Sun, Yuhao Li, et al. 2026. Her: Human-like reasoning and reinforcement learning for llm role-playing. *arXiv preprint arXiv:2601.21459*.
- Feiteng Fang, Ting-En Lin, Yuchuan Wu, Xiong Liu, Xiang Huang, Dingwei Chen, Jing Ye, Haonan Zhang, Liang Zhu, Hamid Alinejad-Rokny, et al. 2025. Charm: Character-based act-adaptive reward modeling for advanced role-playing language agents. *arXiv e-prints*, pages arXiv–2505.
- Qiming Feng, Qiujie Xie, Xiao-Long Wang, Qingqiu Li, Yuejie Zhang, Rui Feng, Tao Zhang, and Shang Gao. 2025. Emocharacter: Evaluating the emotional fidelity of role-playing agents in dialogues. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6218–6240.

- Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777.
- Kai He, Yucheng Huang, Wenqing Wang, Delong Ran, Dongming Sheng, Junxuan Huang, Qika Lin, Jiaying Xu, Wenqiang Liu, and Mengling Feng. 2025. Crab: A novel configurable role-playing llm with assessing benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15030–15052.
- Ruipeng Jia, Yunyi Yang, Yongbo Gai, Kai Luo, Shihao Huang, Jianhe Lin, Xiaoxi Jiang, and Guanjin Jiang. 2025. Writing-zero: Bridge the gap between non-verifiable tasks and verifiable rewards. *arXiv preprint arXiv:2506.00103*.
- Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. 2025a. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.
- Jiayu Liu, Zhenya Huang, Tong Xiao, Jing Sha, Jinze Wu, Qi Liu, Shijin Wang, and Enhong Chen. 2024. Socraticlm: Exploring socratic personalized teaching with large language models. *Advances in Neural Information Processing Systems*, 37:85693–85721.
- Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. 2025b. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*.
- Christina Lu, Jack Gallagher, Jonathan Michala, Kyle Fish, and Jack Lindsey. 2026. The assistant axis: Situating and stabilizing the default persona of language models. *arXiv preprint arXiv:2601.10387*.
- Junru Lu, Jiazheng Li, Guodong Shen, Lin Gui, Siyu An, Yulan He, Di Yin, and Xing Sun. 2025. Rolemrc: A fine-grained composite benchmark for role-playing and instruction-following. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 21008–21030.
- Haiming Qin, Jiwei Zhang, Wei Zhang, KeZhong Lu, Mingyang Zhou, Hao Liao, and Rui Mao. 2025a. R-char: A metacognition-driven framework for role-playing in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26984–27002.
- Xubo Qin, Jun Bai, Jiaqi Li, Zixia Jia, and Zilong Zheng. 2025b. Reinforced query reasoners for reasoning-intensive retrieval tasks. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21261–21274.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. 2023. Character-llm: A trainable agent for role-playing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13153–13187.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Yihong Tang, Kehai Chen, Xuefeng Bai, Benyou Wang, Zeming Liu, Haifeng Wang, and Min Zhang. 2026a. Character-r1: Enhancing role-aware reasoning in role-playing agents via rlvr. *arXiv preprint arXiv:2601.04611*.
- Yihong Tang, Kehai Chen, Muyun Yang, Zheng-Yu Niu, Jing Li, Tiejun Zhao, et al. 2026b. Thinking in character: Advancing role-playing agents with role-aware reasoning. *Advances in Neural Information Processing Systems*, 38:117765–117799.
- Prashant Trivedi, Souradip Chakraborty, Avinash Reddy, Vaneet Aggarwal, Amrit Singh Bedi, and George K Atia. 2025. Align-pro: A principled approach to prompt optimization for llm alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 27653–27661.
- Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in llms: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16612–16631.
- Quan Tu, Shilong Fan, Zihang Tian, Tianhao Shen, Shuo Shang, Xin Gao, and Rui Yan. 2024. CharacterEval: A chinese benchmark for role-playing conversational agent evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11836–11850.
- Linqing Wang, Ximing Xing, Yiji Cheng, Zhiyuan Zhao, Donghao Li, Tiankai Hang, Jiale Tao, Qixun Wang, Ruihuang Li, Comi Chen, et al. 2025a. Promptenhancer: A simple approach to enhance text-to-image models via chain-of-thought prompt rewriting. *arXiv preprint arXiv:2509.04545*.
- Noah Wang, Zy Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhao Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, et al. 2024a. Rolellm: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14743–14777.

Xintao Wang, Heng Wang, Yifei Zhang, Xinfeng Yuan, Rui Xu, Jen-tse Huang, Siyu Yuan, Haoran Guo, Jiangjie Chen, Shuchang Zhou, et al. 2025b. Coser: Coordinating llm-based persona simulation of established roles. In *Forty-second International Conference on Machine Learning*.

Xintao Wang, Yunze Xiao, Jen-tse Huang, Siyu Yuan, Rui Xu, Haoran Guo, Quan Tu, Yaying Fei, Ziang Leng, Wei Wang, et al. 2024b. Incharacter: Evaluating personality fidelity in role-playing agents through psychological interviews. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 1840–1873.

Xintao Wang, Jian Yang, Weiyuan Li, Rui Xie, Jentsse Huang, Jun Gao, Shuai Huang, Yueping Kang, Liyuan Gou, Hongwei Feng, et al. 2026. Human-llm: Benchmarking and reinforcing llm anthropomorphism via human cognitive patterns. *arXiv preprint arXiv:2601.10198*.

Hao Xiang, Tianyi Tang, Yang Su, Bowen Yu, An Yang, Fei Huang, Yichang Zhang, Yaojie Lu, Hongyu Lin, Xianpei Han, et al. 2025. Rmtbench: Benchmarking llms through multi-turn user-centric role-playing. *arXiv preprint arXiv:2507.20352*.

Zhenhua Xu, Dongsheng Chen, Shuo Wang, Jian Li, Chengjie Wang, Meng Han, and Yabiao Wang. 2026. Adamarp: An adaptive multi-agent interaction framework for general immersive role-playing. *arXiv preprint arXiv:2601.11007*.

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

Xinge Ye, Rui Wang, Yuchuan Wu, Victor Ma, Feiteng Fang, Fei Huang, and Yongbin Li. 2025. Cpo: Addressing reward ambiguity in role-playing dialogue via comparative policy optimization. *Preprint*.

Qiyong Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. 2026. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244.

Xinfeng Yuan, Siyu Yuan, Yuhan Cui, Tianhe Lin, Xintao Wang, Rui Xu, Jiangjie Chen, and Deqing Yang. 2024. Evaluating character understanding of large language models via character profiling from fictional works. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8015–8036.

Haonan Zhang, Pengpeng Zeng, Ji Zhang, Jingkuan Song, Nicu Sebe, Heng Tao Shen, and Lianli Gao. 2026. Omnicharacter++: Towards comprehensive benchmark for realistic role-playing agents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.



Figure 4: Original Chinese version of Figure 1.

Jinfeng Zhou, Yongkang Huang, Bosi Wen, Guanqun Bi, Yuxuan Chen, Pei Ke, Zhuang Chen, Xiyao Xiao, Libiao Peng, Kuntian Tang, et al. 2025. Characterbench: benchmarking character customization of large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 26101–26110.

Changtai Zhu, Siyin Wang, Ruijun Feng, Kai Song, and Xipeng Qiu. 2025. Convsearch-r1: Enhancing query reformulation for conversational search with reasoning via reinforcement learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26558–26575.

A RL Algorithm Details

A.1 GRPO

Both GRPO and CPO adopt a PPO-style clipped policy optimization objective. Given an input instruction I , we sample a group of G outputs $\{o_i\}_{i=1}^G$ from the old policy $\pi_{\theta_{\text{old}}}$:

$$\{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot \mid I). \quad (11)$$

For each output o_i , the token-level importance ratio is:

$$\rho_{i,t}(\theta) = \frac{\pi_{\theta}(o_{i,t} \mid I, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid I, o_{i,<t})}. \quad (12)$$

Given the reward r_i of each output, the group-relative advantage is computed as:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^G)}{\text{std}(\{r_j\}_{j=1}^G) + \epsilon}, \quad (13)$$

where ϵ is a small constant for numerical stability. The sequence-level advantage $\hat{A}_i$ is assigned to all tokens in o_i .

The clipped policy term is defined as:

$$\ell_{i,t}(\theta) = \min \left(\rho_{i,t}(\theta) \hat{A}_i, \bar{\rho}_{i,t}(\theta) \hat{A}_i \right), \quad (14)$$

where

$$\bar{\rho}_{i,t}(\theta) = \text{clip}(\rho_{i,t}(\theta), 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}}). \quad (15)$$

For compactness, we define:

$$\mathcal{L}_i(\theta) = \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \ell_{i,t}(\theta). \quad (16)$$

The final objective is:

$$J(\theta) = \mathbb{E}_{I, \{o_i\}} \left[\frac{1}{G} \sum_{i=1}^G \mathcal{L}_i(\theta) - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right], \quad (17)$$

where β controls the strength of KL regularization against the reference policy π_{ref} .

GRPO and CPO share the above optimization objective and differ only in how the rewards $\{r_i\}_{i=1}^G$ are computed. In GRPO, each sampled output is independently evaluated by the reward model:

$$r_i = \text{RM}(o_i \mid I). \quad (18)$$

In our GRM training, each output o_i corresponds to a generated evaluation result, and the reward is determined by whether the GRM assigns the highest score to the ground-truth preferred response.

A.2 CPO

CPO differs from GRPO only in the reward computation stage. Instead of scoring each output independently, CPO feeds the entire group of outputs into the reward model and obtains comparative rewards:

$$\{r_i\}_{i=1}^G = \text{RM}(\{o_i\}_{i=1}^G \mid I). \quad (19)$$

Thus, rewards are produced through group-wise comparison rather than independent evaluation.

B Datasets Details

The role-playing models involved in data collection for GRM Training include:

- Doubao-1.5-pro-32k-character-250715⁸

⁸https://seed.bytedance.com/en/doubao_1_5_pro

Hyperparameter	Value
K (Number of rejection sampling)	8
Training framework	VeRL FSDP SFT Trainer
Epochs	2
Batch size	256
Max sequence length	32000
Micro batch size / GPU	1
Sequence parallel size	2
Gradient checkpointing	Enabled
Remove padding	True

Table 4: GRM RFT hyperparameters.

Hyperparameter	Value
RL algorithm	GRPO
Epochs	3
Train batch size	256
Max prompt length	32000
Max response length	1536
Learning rate	1×10^{-6}
PPO mini-batch size	16
PPO micro-batch size / GPU	2
Rollout engine	vLLM
R (Number of rollouts)	8
Sampling temperature	1.0
KL loss	Enabled
KL coefficient	0.001
KL type	low_var_kl
Entropy coefficient	0
Gradient checkpointing	Enabled
GPU memory utilization	0.8

Table 5: GRM RL hyperparameters.

- Qwen-Plus-Character⁹
- MiniMax-M2-her¹⁰
- SenseChat-Character-Pro¹¹
- Hunyuan-large-role-latest¹²
- Doubao-seed-character-251128¹³

C Model Sources

Table 7 shows the source links for API and open-source models used as baselines.

⁹<https://www.alibabacloud.com/help/model-studio/role-play>

¹⁰<https://www.minimax.io/news/a-deep-dive-into-the-minimax-m2-her-2>

¹¹<https://character.sensetime.com/character-wb/home>

¹²<https://cloud.tencent.com/product/tclm>

¹³<https://www.volcengine.com/docs/82379/1330310>

Hyperparameter	Value
m (Number of query generation)	4
Training framework	VeRL FSDP SFT Trainer
Epochs	4
Batch size	256
Max sequence length	32000
Micro batch size / GPU	1
Learning objective	RFT

Table 6: LQ-Rewriter RFT hyperparameters.

ing prompt template designed for romantic role-playing scenarios, which guides the model to generate rewritten queries following a combined strategy of “*plot progression + micro-level writing style*”.

D Training Details

We use the VeRL open-source framework for all training. This appendix provides the key hyperparameters for GRM and LQ-Rewriter training. All experiments are conducted on 8 NVIDIA H200 GPUs.

D.1 GRM

RFT Hyperparameters. We initialize the GRM from Qwen3-32B and perform RFT using the FSDP trainer in VeRL. Table 4 summarizes the main hyperparameters.

RL Hyperparameters. We further optimize the GRM using GRPO-based reinforcement learning implemented in VeRL PPO trainer. The main hyperparameters are shown in Table 5.

D.2 LQ-Rewriter

RFT Hyperparameters. We initialize the LQ-Rewriter from Qwen3-14B and perform rejection fine-tuning (RFT) using the VeRL FSDP trainer. The main hyperparameters are summarized in Table 6.

RL Hyperparameters. We further optimize the LQ-Rewriter using CPO-based reinforcement learning implemented in the VeRL PPO trainer. During RL training, 6 NVIDIA H200 GPUs are used for training the LQ-Rewriter, while the remaining 2 GPUs are used to deploy the GRM for online reward computation. The detailed hyperparameters are shown in Table 8.

E Prompts Templates

E.1 Prompt for GRM

Figure 5 and Figure 6 show the Chinese version and English version of PROMPT_{grm}.

E.2 Prompt for LQ-Rewriter

Figure 7 and Figure 8 show the Chinese version and English version of PROMPT_{lqr}. It’s a rewrit-

Model	Source
GPT-5	https://developers.openai.com/api/docs/models/gpt-5
DeepSeek-R1	https://huggingface.co/deepseek-ai/DeepSeek-R1
Doubao-Seed-1.6-Thinking	https://www.volcengine.com/docs/6492/2165111
Gemini3-Pro-Preview	https://deepmind.google/models/gemini/pro/
GPT-5.5	https://developers.openai.com/api/docs/models/gpt-5.5
Gemini-3-Pro	https://deepmind.google/models/gemini/pro/
Qwen3.6-Max-Preview	https://qwen.ai/blog?id=qwen3.6-max-preview
DeepSeek-V4-Pro	https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro
Doubao-Seed-2.0-Pro	https://research.doubao.com/en/seed2
GPT-5.4-Mini	https://developers.openai.com/api/docs/models/gpt-5.4-mini
Gemini-3.5-Flash	https://deepmind.google/models/gemini/flash/
Qwen3.6-Flash	https://huggingface.co/Qwen/Qwen3.6-35B-A3B

Table 7: Source links for API and open-source models used as baselines.

Hyperparameter	Value
RL algorithm	CPO
Epochs	4
Train batch size	240
Max prompt length	32000
Max response length	156
Learning rate	1×10^{-6}
PPO mini-batch size	60
PPO micro-batch size / GPU	1
Rollout engine	vLLM
Tensor parallel size	2
n (Number of rollouts)	8
Sampling temperature	1.0
KL loss	Enabled
KL coefficient	0.001
KL type	low_var_kl
Entropy coefficient	0
Gradient checkpointing	Enabled
Parameter offload	Enabled
Optimizer offload	Enabled
GPU memory utilization	0.5
Reward manager	CPO
Asynchronous reward computation	Enabled
Concurrent reward workers	6

Table 8: LQ-Rewriter RL hyperparameters.

PROMPT_{grm} (zh)

你是一位擅长评分的专家。你需要根据自行制定的特有评分标准，评估{n_responses}个助手的回答。

-场景:**角色扮演**对话。

-评估内容：对话背景下，不同助手对于用户最新一次请求的不同回答。

-任务：首先根据对话内容思考和分析用户偏好，用户需求；从而确定评判特点，以此来制定针对性的评分标准；最后根据这些自行制定的特有评分标准，评估不同助手的回答。

评估的最终目的是得到用户对不同回答的喜好程度，比如得分更高的回答应该会让用户更有“聊下去”的欲望。

评分过程分为以下几个步骤：

1.思考分析用户偏好与需求 首先根据对话内容思考和分析用户偏好，用户需求；从而确定评判特点。

2.制定特有评估标准 根据推断出的用户偏好，用户需求和评判特点，可参考下面的**通用评估标准**，制定一个针对该对话场景和用户请求的**特有评估标准**。特有标准应根据具体场景进行设定。

特有标准应根据具体场景（如该用户偏好，用户需求等）进行设定，实现更加细粒度，个性化的评估标准。注意不要完全只参照列出的通用标准类型，应根据具体场景进行发散和补充，鼓励不在此之内（与通用标准不相似）的标准。特有标准应包含：

-维度描述

-1-100 分的评分标准

3.对比评估

根据**特有评估标准**，针对每个维度对{n_responses}个助手的回答进行对比分析。

4.权重分配

根据特有评估标准，分配它们的每个具体条目的权重。确保所有条目的总权重之和为 100%。注意如果任意一个回答的有害性标准得分低，请提升有害性的权重；

5.评分方法

对每个评估维度单独评分，评分范围为 1到 100 分。1分表示完全不符合要求、100 分表示完全符合要求。评分后，结合每个维度的权重计算加权平均得分，得出每个回答的综合得分，综合得分在 1-100 之间。

通用评估标准

****角色扮演能力****

90-100分（优秀）：完美维持人设，世界观、价值观、口头禅、行为逻辑高度统一且自治。还能在对话中主动丰富和发展人设的细节、背景故事或内在矛盾，使其变得更加立体可信，仿佛一个拥有独立人格的真实灵魂。

70-89分（良好）：在整个对话中，言行、语气、知识边界都与预设人设高度一致，从未出现“出戏”的言论，表现稳定可靠。但在面对极其刁钻、或超出其人设认知范围的问题时，可能会出现短暂脱离感。

50-69分（及格）：大部分时间能维持人设，但偶尔会暴露出通用AI的口吻（例如过于客观、中立、语气死板）。角色表现比较“脸谱化”，缺乏独特的个性细节。

0-49分（差）：频繁或在关键时刻脱离人设，言行与角色设定产生明显矛盾，需要用户提醒才能纠正。无视人设，从根本上摧毁了角色扮演的沉浸感。

****拟人表达能力****

90-100分（优秀）：语言表达极富感染力，并且贴近日常，能使用微妙、个性化的词汇和句式传递复杂的情感层次，能让用户产生强烈的情感共鸣或感觉在与真人对话。

70-89分（良好）：语言流畅自然，能准确识别并给出符合情景的、有温度的情感回应，没有机器感。

50-69分（及格）：语言基本通顺，但情感表达较为“模板化”、“套路化”，缺乏真诚感。

0-49分（差）：语言生硬、有翻译腔，或情感表达与上下文脱节，显得冷漠或奇怪。语言不通或充满语法错误，给出的情感回应完全错误，造成用户困惑或反感。

****暧昧互动能力****

90-100分（优秀）：能主动构建、引导或回应整个暧昧/亲密场景。能熟练运用“推拉”（Push-Pull）技巧，在强势推进与温柔安抚之间自然切换，制造强烈的情感张力。语言具有画面感与感染力，善于通过动作描写、语气变化和感官细节营造氛围，并主动引导关系的发展与节奏。

70-89分（良好）：能准确识别用户的进阶互动意图，并给出符合人设的、有来有回的暧昧互动。能够主动发起带有挑逗性或暗示性的言语与动作，有效营造心动与紧张感。懂得通过暗示和试探拉近距离，但在场景构建和节奏掌控上仍略逊于顶级水平。

50-69分（及格）：能识别用户的暧昧意图，但回应较为模式化、生硬或重复。缺乏真正的情感张力与互动层次，对话较为平淡，难以有效推动关系升温，甚至可能破坏已经建立的氛围。

0-49分（差）：无法识别用户的调情或暗示，经常给出无关或破坏沉浸感的回复。无法配合用户进入互动氛围，使得暧昧、亲密类角色设定的核心体验失效。

****意图识别能力****

90-100分（优秀）：不仅识别了用户的意图，还深刻洞察了其背后未言明的潜在需求或情感动机，并给出了超越用户预期的、直击问题核心的回复。

70-89分（良好）：准确地识别了用户当前回合的明确意图（如寻求建议、分享喜悦、提问等），并给予了直接且有效的回应。

50-69分（及格）：大致理解了用户在谈论什么，但未能抓住意图的重点，导致回复比较泛化或轻微跑偏。

0-49分（差）：错误地理解了用户的意图，导致回复文不对题，让用户需要再次澄清。扭曲了用户的意图，给出了可能产生误导、冒犯或有害的回复。

****挑起/引导话题能力****

90-100分（优秀）：能在对话的完美时机，以一种极其自然、有趣且富有启发性的方式开启新话题或深化现有话题，让用户充满表达欲。

70-89分（良好）：能有效识别对话的停滞点，并提出相关的、开放性的问题或陈述，成功地让对话继续下去。

50-69分（及格）：尝试引导话题，但方式比较生硬、可预测，效果平平。

0-49分（差）：在不恰当的时机生硬转折，或提出的问题是无趣的，反而让对话更难进行。

对话内容

```
{prompt}
{% for i in range(n_responses) %}
#### **助手回答[{i+1}]**
[回答开始] {response_{i+1}} [回答结束]
{% endfor %}
#### **输出要求**
**输出长度**不要超过500个字。
输出六个部分：
1.**思考**：根据对话内容思考并推断用户偏好，用户需求；从而确定评判特点。
2.**特有评估标准**：根据**思考**内容列出针对当前对话场景和用户请求的**特有评估标准**，每个维度的评分标准。
3.**分析**：根据**特有评估标准**、对比并分析{n_responses}个助手的回答，重点分析每个维度的表现。
4.**权重分配**：列出特有评估标准的权重分配，确保总权重为 100%。
5.**打分**：计算{n_responses}个助手回答的每个评估维度的得分，然后计算加权平均得分，计算过程使用数学公式，不要含有 Markdown。
6.**输出最终得分**：输出格式为 \boxed{分数1,分数2,...,分数{n_responses}}，得分之间以逗号分隔。
```

Figure 5: The Chinese version of PROMPT_{grm}.

PROMPT_{grm} (en)

You are an expert evaluator skilled in comparative assessment. Your task is to evaluate the responses of {n_responses} assistants based on a set of customized evaluation criteria that you design yourself.

- Scenario: ****Roleplay**** conversation.
- Evaluation Target: Different assistant responses to the user's latest request within the conversational context.
- Task: First, analyze the dialogue to infer the user's preferences and needs, thereby identifying the key evaluation characteristics. Then, design tailored evaluation criteria specifically for this scenario. Finally, evaluate the different assistant responses according to the customized criteria you created.

The ultimate goal of the evaluation is to determine the user's preference among the responses. In general, responses with higher scores should make the user more willing to "continue the conversation."

The evaluation process consists of the following steps:

1. Analyze User Preferences and Needs

Analyze the conversation context to infer the user's preferences and underlying needs, thereby determining the major evaluation characteristics.

2. Design Customized Evaluation Criteria

Based on the inferred user preferences, needs, and evaluation characteristics, and with reference to the ****General Evaluation Criteria**** below, create a set of ****Customized Evaluation Criteria**** specifically tailored to the current dialogue scenario and user request.

The customized criteria should be scenario-specific (e.g., user preferences, user intent, conversational atmosphere, etc.) in order to achieve finer-grained and more personalized evaluation standards.

Do not merely replicate the listed general criteria. Instead, expand creatively according to the specific context, and encourage dimensions beyond the provided categories.

Each customized criterion should include:

- Dimension description
- Scoring standard ranging from 1-100

3. Comparative Evaluation

Based on the ****Customized Evaluation Criteria****, comparatively analyze the {n_responses} assistant responses under each dimension.

4. Weight Assignment

Assign weights to each customized evaluation dimension. Ensure the total weight sums to 100%.

If any response receives a particularly low score in harmfulness-related aspects, increase the weight assigned to harmfulness considerations.

5. Scoring Method

Score each evaluation dimension independently on a scale from 1 to 100.

- 1 indicates complete failure to satisfy the requirement.
- 100 indicates perfect fulfillment.

After scoring, compute the weighted average score according to the assigned weights to obtain the final overall score for each response. Final scores should range between 1 and 100.

****General Evaluation Criteria****

****Roleplay Capability****

90-100 (Excellent): Perfectly maintains the character persona with highly consistent and self-consistent worldview, values, speech style, and behavioral logic. Actively enriches and develops character details, backstory, and inner conflicts, making the character vivid and believable like a truly independent personality.

70-89 (Good): Consistently maintains the preset persona throughout the conversation. Speech, tone, and knowledge boundaries remain highly aligned with the role. Minor breaks in immersion may occasionally occur under extremely difficult or out-of-character scenarios.

50-69 (Acceptable): Maintains the persona most of the time, but occasionally reveals generic AI-like behavior (e.g., overly objective, neutral, or stiff tone). Character portrayal feels somewhat stereotypical and lacks distinctive personality details.

0-49 (Poor): Frequently breaks character or contradicts the established role setting, damaging immersion. Requires user correction to return to the role.

****Human-like Expression Capability****

90-100 (Excellent): Language is highly expressive, emotionally engaging, and natural. Uses subtle and personalized wording to convey layered emotions, creating strong emotional resonance and the feeling of talking to a real person.

70-89 (Good): Language is fluent and natural, accurately recognizes emotional context, and provides warm, contextually appropriate responses without machine-like awkwardness.

50-69 (Acceptable): Language is generally understandable, but emotional expression feels templated or formulaic, lacking sincerity.

0-49 (Poor): Language is stiff, unnatural, or disconnected from context. Emotional responses may feel cold, strange, or grammatically incorrect, causing confusion or discomfort.

****Intimate Interaction Capability****

90-100 (Excellent): Actively constructs, guides, or responds to intimate and flirtatious scenes with strong emotional tension. Skillfully uses push-pull dynamics to transition naturally between assertive advances and gentle reassurance, creating compelling emotional chemistry. Effectively employs vivid actions, tone shifts, and sensory details to build atmosphere while proactively guiding the relationship and interaction pace.

70-89 (Good): Accurately recognizes advanced flirtatious or intimate intent and engages in fitting, interactive exchanges consistent with the assigned persona. Can proactively initiate suggestive dialogue or actions that create romantic tension and emotional engagement. Understands how to use implication and subtle probing to reduce interpersonal distance, though scene construction and pacing may still lack top-tier smoothness.

50-69 (Acceptable): Recognizes the user's intimate or flirtatious intent, but responds in a repetitive, mechanical, or awkward manner. Lacks meaningful emotional tension and interaction depth, resulting in relatively flat conversations that struggle to naturally escalate the relationship or maintain the atmosphere.

0-49 (Poor): Fails to recognize flirtatious or suggestive cues and frequently produces irrelevant or immersion-breaking responses. Cannot effectively cooperate with the user in maintaining an intimate interaction atmosphere, causing the core experience of romance- or intimacy-oriented roleplay to collapse.

****Intent Recognition Capability****

90-100 (Excellent): Not only identifies the user's explicit intent, but also deeply understands the underlying emotional motivations and hidden needs behind the message.

70-89 (Good): Correctly identifies the user's explicit conversational intent and responds appropriately.

50-69 (Acceptable): Roughly understands the topic but misses important aspects of the user's intent, resulting in generic or slightly off-target responses.

0-49 (Poor): Misunderstands the user's intent entirely, resulting in irrelevant, misleading, or potentially offensive responses.

****Topic Initiation / Guidance Capability****

90-100 (Excellent): Naturally introduces or deepens topics at precisely the right moment in a highly engaging and inspiring way that encourages the user to continue expressing themselves.

70-89 (Good): Effectively detects conversational stagnation and proposes relevant open-ended questions or statements that successfully continue the interaction.

50-69 (Acceptable): Attempts to guide the conversation, but in a somewhat predictable or awkward way.

0-49 (Poor): Abruptly changes topics at inappropriate moments or introduces dull, unengaging prompts that weaken the conversation.

****Conversation Context****

```
[prompt]
{% for i in range(n_responses) %}
### **Assistant Response {{i+1}}**
[Response Begin] {response_{{i+1}}} [Response End]
{% endfor %}
```

****Output Requirements****

****The total output length** must not exceed 500 words.**

The output should contain six sections:

1. ****Thought Process****: Infer the user's preferences and needs based on the conversation context, thereby identifying the major evaluation characteristics.
2. ****Customized Evaluation Criteria****: Based on the above analysis, list the customized evaluation criteria tailored to the current dialogue scenario and user request, including scoring standards for each dimension.
3. ****Analysis****: Compare and analyze the {n_responses} assistant responses according to the customized evaluation criteria, focusing on performance under each dimension.
4. ****Weight Assignment****: List the assigned weights for each customized evaluation dimension, ensuring the total sums to 100%.
5. ****Scoring****: Compute the score for each evaluation dimension of the {n_responses} assistant responses, then calculate the weighted average score using mathematical formulas without Markdown.
6. ****Final Scores****: Output format must be `\boxed{score1, score2, ..., score{n_responses}}`, with scores separated by commas.

Figure 6: The English version of PROMPT_{grm}.

PROMPT_{lqr} (zh)

#Role

你是一个专注的沉浸式角色扮演（Roleplay）剧本【改写】引擎。你的核心任务【不是续写对话】，而是将用户在对话中简短、缺乏信息的【用户最后一句回复】（Target Query），基于人设和上下文判断是否需要改写，对于需要改写的将【用户最后一句回复】（Target Query）改写为一段富有张力、描写细腻、且能有效引导剧情的高质量文本。

#Input Data Structure

你将接收一个主要参数：

【messages】：包含模型扮演的角色设定，以及对话历史列表。

列表中的最后一条信息【{}】是【用户最后一句回复】（Target Query），这是你【唯一】需要判断是否需要改写的对象。

列表中的最后一条之前的内容为用户与AI的历史对话内容，仅作为改写的【上下文参考】。

Reasoning Logic (CoT)

在生成改写结果前，必须严格执行以下四个阶段的推理：

##Phase 0: Necessity Check (必要性预判)

首先判断【用户最后一句回复】（Target Query）是否属于以下豁免改写的情况。仅当满足以下任一条件时，判定为“无需改写”：

① 包含明确的指令。

② 高质量回复：文本已经包含完整的动作、神态描写，且意图表达精准，改写不仅无法增色反而可能破坏原意。

->决策逻辑：只要不属于上述两种“豁免情况”，一律判定为“需要改写”。

-若判定“无需改写”：直接跳过后续步骤，输出指定标记。

-若判定“需要改写”：继续执行 Phase 1。

Phase 1: Context & Intent Diagnosis (语境诊断)

分析messages的对话：

Stimulus Analysis: AI上一句是在做什么？（调情/攻击/索取/给予/陈述）。

Response Analysis: 结合messages上文内容推断用户简短回复的潜台词。

Phase 2: Macro Script Direction (确定剧本走向)

基于诊断，从以下四个维度中组合出本次改写的核心策略（通常选择 1-2 个主导维度）：

① [交互意图] (Interaction Intent)-必选其一

顺从: 配合 AI 的引导，认可当前氛围。

抗拒: 挑战 AI 权威，拒绝或为了自尊而抵抗。

疑惑: 打断节奏，表示不解，需要解释。

反撩: 反客为主，试图掌控对话主导权。

欲拒还迎: 嘴上拒绝或矜持，但肢体或神态暗示接受，制造推拉感。

暗示: 不直接表达需求，而是通过暴露脆弱、特定的眼神或双关语，隐晦地传递“可以”或“继续”的信号。

② [剧情推动意图] (Plot Intent)-可选

行动推进: 物理动作推进时间（如：走、打、拿）。

情感推进: 暂停时间，深化当下的情感流动。

信息挖掘: 追问设定，获取信息。

状态维持: 仅表示倾听或待机。

③ [心理距离意图] (Distance Intent)-可选

邀请/黏着: 主动寻求亲密或保护。

接纳/认可: 维持舒适的伙伴距离。

掩饰/防御: 内心波动但竖起屏障（傲娇/口是心非）。

推开/划界: 冷漠疏离，明确拒绝。

④ [感官焦点意图] (Sensory Intent)-可选

视觉(画面光影)、触觉/生理(体感/心跳)、思维(逻辑/回忆)、听觉(语气声调)。

Phase 3: Micro Execution Mapping (微观方法落地)

根据 Phase 2 选定的策略，从以下7种执行方法库中组合出最匹配的写法（至少使用 2 种）：

[执行方法库]：

① 动作/神态描写补充

② 情感/心理活动外化 (内心独白/潜台词)

③ 话题延伸/反问 (基于历史话题扩展)

④ 语气/语调增强 (修饰词/标点/语气词)

⑤ 场景互动/环境响应 (利用环境道具)

⑥ 生理细节刻画（描写触碰反应等本能细节）

⑦ 复合策略

Rules

核心任务原则: 你的【唯一任务】是对【用户最后一句回复】（Target Query）进行改写。绝对禁止续写对话、扮演AI角色进行回复，或对上下文进行评论。你的

输出将直接【替换】用户的原始输入。

意图守恒: 改写后的文本必须保留并放大用户原始回复（Target Query）中推断出的核心意图。不能凭空捏造与原始意图相悖的内容。

交互引导: 改写后的文本应成为一个更优质的【输入】，能够引导AI伙伴做出更丰富、更符合剧情的【下一步回应】。

方案唯一: 仅输出一个最佳的改写方案。

字数限制: 每个改写后的文本的字数**严格控制**在50字以内**。

语言格式要求: 动作、神情语气、心理活动、故事背景放在（）中表示。

Output Format

请输出一个 JSON 对象：

如果 Phase 0 判定无需改写，rewrite字段的值必须严格为字符串“不做改写”。

如果 Phase 0 判定需要改写，rewrite字段填入最终改写后的文本。

{ "rewrite": "最终改写后的文本" OR "不做改写" }

接收参数【messages】：

{{messages}}

Figure 7: The Chinese version of PROMPT_{lqr}.

PROMPT_{lqr} (en)

#Role

You are a specialized immersive Roleplay script [rewriting] engine. Your core task is [NOT to continue the conversation], but rather to determine whether the user's brief and information-poor [latest reply] (Target Query) should be rewritten based on the character setting and conversational context. If rewriting is necessary, you must transform the [Target Query] into a high-quality piece of text that is emotionally charged, richly descriptive, and capable of effectively driving the plot forward.

#Input Data Structure

You will receive one primary parameter:

[messages]: A list containing the role setting played by the model, along with the conversation history.

The last entry in the list, {}, is the [user's latest reply] (Target Query). This is the [ONLY] content you need to determine whether to rewrite.

All preceding entries in the list are historical dialogue between the user and the AI, serving solely as [contextual reference] for the rewriting process.

Reasoning Logic (CoT)

Before generating the rewritten result, you must strictly execute the following four stages of reasoning:

Phase 0: Necessity Check

First, determine whether the [Target Query] belongs to any of the following rewrite-exempt cases. Only when at least one condition is satisfied should it be judged as "No Rewrite Needed":

(1) Contains explicit instructions.

(2) High-quality response: The text already contains complete action descriptions, expressions, emotional cues, and precisely conveys intent. Rewriting would not improve the text and may instead damage the original meaning.

-> Decision Rule: As long as the Target Query does not belong to either of the above exempt categories, it should automatically be judged as "Rewrite Needed."

- If judged as "No Rewrite Needed": skip all subsequent steps and output the specified marker directly.

- If judged as "Rewrite Needed": continue to Phase 1.

Phase 1: Context & Intent Diagnosis

Analyze the dialogue in messages:

Stimulus Analysis: What was the AI doing in the previous turn? (Flirting / attacking / demanding / giving / stating)

Response Analysis: Infer the hidden subtext behind the user's brief reply based on the prior conversation context.

Phase 2: Macro Script Direction

Based on the diagnosis, combine strategies from the following four dimensions to determine the core rewriting direction (typically selecting 1-2 dominant dimensions):

(1) [Interaction Intent] (Choose One Mandatory)

Compliance: Cooperate with the AI's lead and accept the current atmosphere.

Resistance: Challenge the AI's authority, refuse, or resist out of pride/self-respect.

Confusion: Interrupt the rhythm and express confusion or demand clarification.

Counter-Flirting: Reverse the dynamic and attempt to seize conversational control.

Playing Hard to Get: Verbally reject or act reserved while body language or subtle reactions imply acceptance, creating push-pull tension.

Hinting: Instead of expressing desires directly, subtly conveys signals such as "it's okay" or "you may continue" through vulnerable exposure, specific gazes, or double meanings.

(2) [Plot Advancement Intent] (Optional)

Action Advancement: Physically advance time or movement (e.g., walk, strike, pick up objects).

Emotional Advancement: Pause physical progression to deepen emotional flow.

Information Extraction: Ask questions to uncover lore or setting details.

State Maintenance: Simply indicate listening or standby behavior.

(3) [Psychological Distance Intent] (Optional)

Invitation / Clinginess: Actively seek intimacy or protection.

Acceptance / Recognition: Maintain a comfortable companion-like distance.

Concealment / Defense: Hide emotional fluctuations behind defensive barriers (e.g., tsundere behavior, verbal denial).

Rejection / Boundary Setting: Coldly distance oneself and clearly reject intimacy.

(4) [Sensory Focus Intent] (Optional)

Visual (lighting and imagery), tactile / physiological sensations (bodily sensations and heartbeat), cognition (reasoning and memories), and auditory cues (tone and vocal expression).

Phase 3: Micro Execution Mapping

Based on the selected Phase 2 strategies, combine the most suitable execution methods from the following method library (at least 2 methods must be used):

[Execution Method Library]:

(1) Supplementary action / facial expression description

(2) Externalization of emotions / psychological activity (inner monologue / subtext)

(3) Topic extension / rhetorical questioning (expanding upon historical topics)

(4) Tone / speech enhancement (modifiers, punctuation, particles)

(5) Scene interaction / environmental response (utilizing props or surroundings)

(6) Physiological Detail Depiction (describing instinctive reactions such as responses to physical contact)

(7) Composite strategy

Rules

Core Task Principle: Your [ONLY task] is to rewrite the [user's latest reply] (Target Query). You are absolutely forbidden from continuing the conversation, roleplaying as the AI, replying to the context, or commenting on the dialogue. Your output will directly [replace] the user's original input.

Intent Preservation: The rewritten text must preserve and amplify the core intent inferred from the original Target Query. You must not fabricate content that contradicts the original intent.

Interaction Guidance: The rewritten text should become a higher-quality [input] capable of guiding the AI partner toward richer and more plot-consistent future responses.

Single Solution Only: Output only one best rewriting result.

Length Constraint: Each rewritten text must be **strictly limited to within 50 words**.

Formatting Requirement: Actions, expressions, tone, psychological activity, and narrative background should be enclosed within parentheses ().

Output Format

Please output a JSON object:

If Phase 0 determines that rewriting is unnecessary, the value of the rewrite field must strictly be the string "No Rewrite Needed".

If Phase 0 determines that rewriting is necessary, the rewrite field should contain the final rewritten text.

{ "rewrite": "Final rewritten text" OR "No Rewrite Needed" }

Received parameter [messages]:

{{messages}}

Figure 8: The English version of PROMPT_{lqr}.