

Adversarial Closed-Loop Curriculum for Evolving Role-Playing Agents

Zheng Zhang¹, Liu Liu², Qi Chai¹, Deheng Ye², Peilin Zhao³, Mao Zheng², Hao Wang¹

¹The Hong Kong University of Science and Technology (Guangzhou),

²Tencent,

³Shanghai Jiao Tong University

Abstract

Role-playing agents based on large language models have been widely applied in areas such as personalized assistance and social simulation. Recent RL methods typically train on a fixed scenario pool collected before learning begins. This creates a distributional bottleneck: as the agent improves, the scenarios where it performs poorly also change, while the training distribution remains static. Therefore, we propose AdvRole, an adversarial context rewriting framework that turns role-playing RL into a closed-loop curriculum. AdvRole alternates between an Actor that learns to role-play and a Rewriter that edits character profiles and dialogue contexts into actor-specific hard scenarios. The Rewriter is trained with a performance-gap reward, which favors rewrites that reduce the current Actor’s score relative to the original scenario. As a result, the scenario pool evolves with the Actor and continuously targets under-mastered regions of the character-context space. Experiments on three role-playing benchmarks covering English and Chinese, as well as a new multilingual benchmark we release, show that AdvRole consistently outperforms baselines.

Introduction

Large language models (LLMs) have demonstrated remarkable capabilities in open-ended generation, ranging from creative writing (Chung et al. 2025; Wei et al. 2025; Rashkin et al. 2025; Saakyan et al. 2026) to emotional support (Chen et al. 2023; Zheng et al. 2025; Yuan et al. 2025; Wang et al. 2025a) and communication game (Lan et al. 2024; Zhang et al. 2025a,b). Among these tasks, role-playing has attracted increasing attention (Chen et al. 2024; Wang et al. 2025b, 2026b,a), with applications in interactive entertainment, social simulation, and personalized assistants.

In role-playing, an agent is asked to act as a designated character and produce dialogue that is faithful to the character’s profile and consistent with the ongoing conversation. Early approaches build role-playing agents through prompt engineering (Wang et al. 2024a,b; Han et al. 2024) or supervised fine-tuning on curated dialogue corpora (Shao et al. 2023; Lu et al. 2024; Zhou et al. 2024; Wang et al. 2025b). More recent work turns to reinforcement learning (RL), starting with PPO or DPO on human-annotated preference data (Peng and Shang 2024; Zeng et al. 2025; Ji et al. 2025; Abdulhai et al. 2026), and later moving to GRPO (Shao et al. 2024) using automatically computed rewards that remove the need for human annotation (Liu et al. 2025; Wang et al.

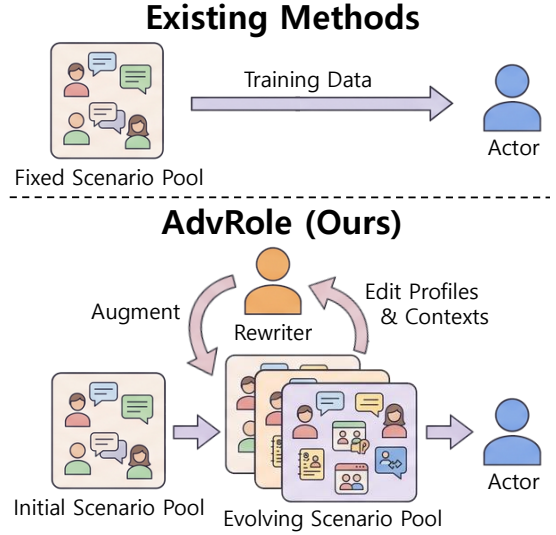


Figure 1: **Comparison between existing methods and AdvRole.** Existing methods train the Actor on a fixed scenario pool collected before training. In contrast, AdvRole introduces a Rewriter that continually edits character profiles and dialogue contexts to expose Actor failures, evolving the scenario pool throughout training.

2025d, 2026b,a). Despite these advances, most RL-based methods remain reward-centric: they improve how responses are scored or optimized, but train on a fixed scenario pool collected before learning begins.

This static training distribution creates a mismatch between optimization and data. A role-playing scenario is the joint product of a character profile and a dialogue context, covering factors such as personality, background, speaking style, scene, partner, and dialogue history. More importantly, scenario difficulty is actor-dependent. As the Actor improves, the scenarios where it still struggles also change. Common profiles and contexts may become uninformative, while useful training data shifts toward rarer traits, subtle conflicts, and unfamiliar dialogue states. Thus, the key question is not only how to reward a response, but also where to train the Actor next. A fixed pool cannot follow this moving failure boundary.

To this end, we propose AdvRole, an adversarial closed-

loop curriculum for evolving role-playing agents. As shown in Figure 1, AdvRole alternates between two models: an **Actor** that learns to perform role-play and a **Rewriter** that edits character profiles and dialogue contexts into actor-specific hard scenarios. The Actor is trained with a standard role-playing reward model. The Rewriter is optimized with a performance-gap reward: a rewritten scenario is considered useful when it causes the current Actor to receive a lower role-playing score than on the original scenario. In this way, the Rewriter acts as a diagnostic adversary, searching for plausible scenario variations that reveal where the Actor currently fails. The resulting scenario pool evolves with the Actor, continuously pushing training toward under-mastered regions of the character-context space.

We evaluate AdvRole on three role-playing benchmarks covering English and Chinese, where it outperforms baselines. To enable broader evaluation, we also release a new multilingual role-playing benchmark, and further demonstrate the effectiveness of AdvRole on it.

Our contributions can be summarized as follows:

- We propose AdvRole, an adversarial RL framework that alternately trains an Actor and a Rewriter, continually improving the Actor’s role-playing ability.
- We introduce a reward based on the Actor’s performance gap that drives the Rewriter to generate more challenging character profiles and contexts for the Actor.
- We demonstrate the effectiveness of our framework on three existing benchmarks, and release a new multilingual role-playing benchmark to support broader evaluation.

Related Work

Role-Playing Agent

Role-playing agents are commonly built by prompting LLMs with character profiles, personality traits, and few-shot dialogues (Wang et al. 2024a,b; Han et al. 2024), or by fine-tuning them on role-playing corpora collected from novels, scripts, and synthetic conversations (Shao et al. 2023; Lu et al. 2024; Zhou et al. 2024; Wang et al. 2025b). More recent work improves role-playing agents with reinforcement learning, using PPO or DPO with human preference data (Ouyang et al. 2022; Rafailov et al. 2023; Peng and Shang 2024; Zeng et al. 2025; Ji et al. 2025; Abdulhai et al. 2026), or GRPO with automatically computed rewards (Shao et al. 2024; Liu et al. 2025; Wang et al. 2025d, 2026b,a). These methods differ in how they elicit, fine-tune, or reward in-character behavior, but they typically optimize the agent on a fixed scenario pool collected before training. AdvRole instead evolves the training distribution itself by adversarially rewriting character profiles and dialogue contexts into actor-specific hard scenarios.

Label-Free Reinforcement Learning

Label-free reinforcement learning (RL) aims to improve LLMs without relying on ground-truth answers or human-annotated preferences. Existing methods derive reward signals from the model’s own behavior, such as sequence-level confidence (Li et al. 2025; Prabhudesai et al. 2025), consistency over multiple reasoning paths (Zuo et al. 2026; Zhang

et al. 2026; Prasad et al. 2025), or output entropy (Agarwal et al. 2026; Cheng et al. 2026). Others train on synthetic tasks created by the model itself, using execution-based or self-derived signals as rewards (Zhao et al. 2026; Zhou et al. 2026). These methods are mostly studied in verifiable domains such as math and code, where automatic verification is straightforward. Role-playing is a natural fit for label-free RL, since there is no single ground-truth response to align against and human preference labels are expensive to obtain at scale.

Self-Play in LLMs

Self-play, where two roles co-evolve through their interaction, has recently emerged as a promising paradigm for improving LLMs without external supervision or human-labeled data. In verifiable domains, a generator and a verifier are paired so that the verifier’s automatic judgment provides reward for the generator, as in coder-tester frameworks for code generation (Lin et al. 2025; Jiang et al. 2025) and proposer-solver frameworks for math reasoning (Zhao et al. 2026; Huang et al. 2026). Our framework follows this self-play paradigm but targets role-playing, where outputs cannot be checked by a deterministic verifier.

Method

Overview

AdvRole trains two models alternately: an **Actor** that performs role-playing, and a **Rewriter** that edits the original character profile and dialogue context to expose the Actor’s failures. Both models are optimized with GRPO (Shao et al. 2024), and the scenario pool for training keeps evolving as the Rewriter improves.

As shown in Figure 2, each epoch of AdvRole consists of three steps: (1) **Actor Training**: the Actor is updated on the current scenario pool using a reward model that scores role-playing quality. (2) **Adversarial Rewriter Training**: the Rewriter is updated with a reward defined by the Actor’s performance gap between the original and rewritten scenarios, encouraging it to expose the weaknesses of the current Actor. (3) **Scenario Augmentation**: the updated Rewriter rewrites the original scenarios into variants that reveal failures, which are merged into the pool to train the Actor in the next epoch.

Actor Training

We formalize a role-playing scenario as a tuple $c = (p, d)$, where p is the character profile and d is the dialogue context that the agent needs to respond to. Given the current scenario pool $\mathcal{S}$, the Actor π_ϕ generates a response conditioned on c :

$$r \sim \pi_\phi(\cdot | c), \quad c \in \mathcal{S}. \quad (1)$$

To assess the role-playing quality of r , we follow Ye et al. (2025) and adopt a strong open-source LLM (Qwen2.5-72B-Instruct¹) as the reward model (RM), which scores r along several role-playing dimensions given the scenario c (the full scoring prompt is provided in the supplementary material):

$$R_{\text{Actor}}(r) = \text{RM}(r | c). \quad (2)$$

¹<https://huggingface.co/Qwen/Qwen2.5-72B-Instruct>

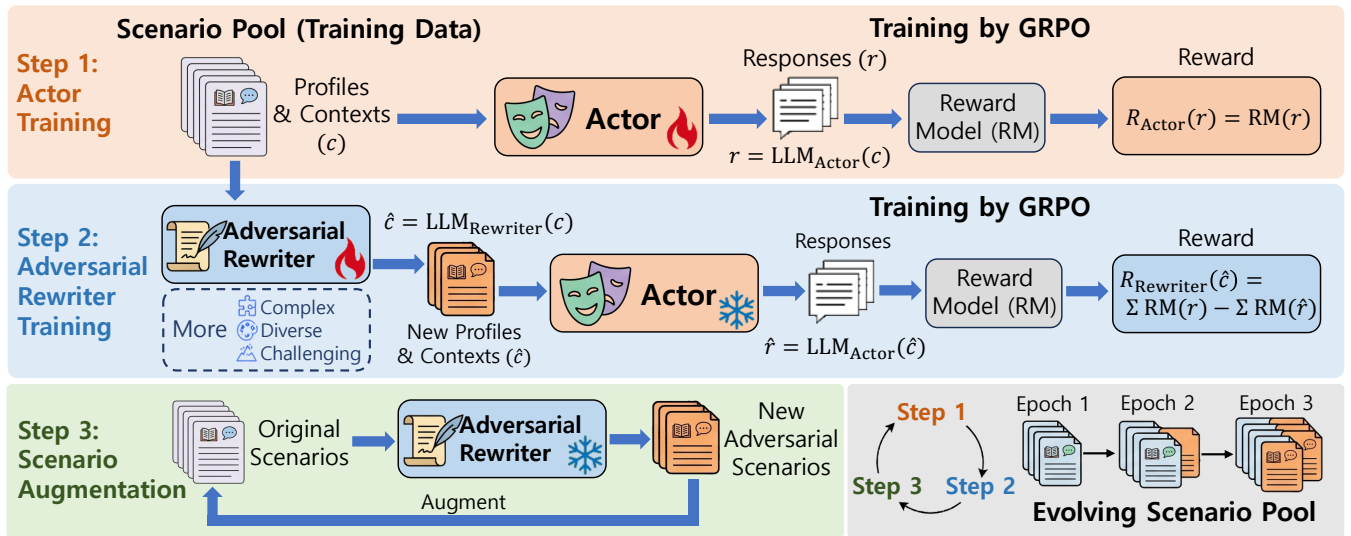


Figure 2: **The training framework of AdvRole.** In each epoch, the Actor is first trained by GRPO on the current scenario pool with rewards from a reward model, and the Rewriter is then trained to edit original scenarios to expose the Actor’s failures, with its reward defined by the Actor’s performance gap between the original and rewritten scenarios. The updated Rewriter further augments the pool for the next epoch, yielding an evolving scenario pool.

To avoid overly long responses, we multiply the Actor reward by a decay coefficient based on the response length $|r|$ (in words for English and in characters for Chinese). The coefficient is 1 below a lower threshold τ_1 , decreases linearly to 0 at an upper threshold τ_2 , and remains 0 beyond τ_2 , with $(\tau_1, \tau_2) = (50, 100)$ for English and $(60, 120)$ for Chinese.

We optimize the Actor with GRPO (Shao et al. 2024). For a policy π_θ trained on inputs x with reward R , GRPO samples a group of G outputs $\{o^{(i)}\}_{i=1}^G \sim \pi_\theta(\cdot | x)$ and maximizes:

$$\mathcal{J}(\theta) = \mathbb{E}_x \left[\frac{1}{G} \sum_{i=1}^G \mathcal{L}_i^\theta(A^{(i)}) - \beta D_{KL}(\pi_\theta || \pi_{\text{ref}}) \right], \quad (3)$$

$$A^{(i)} = \frac{R(o^{(i)}) - \mu}{\sigma},$$

where $\mathcal{L}_i^\theta(A^{(i)})$ is the clipped surrogate objective parameterized by the normalized advantage $A^{(i)}$, and μ, σ are the mean and standard deviation of the group rewards. For the Actor, we instantiate π_θ as π_ϕ , x as the scenario c , and R as R_{Actor} in Equation (2).

Adversarial Rewriter Training

While the Actor is trained on $\mathcal{S}$, the pool itself has limited coverage and diversity. As the Actor gradually fits this pool, further training on the same data brings limited gain. To break this bottleneck, we introduce a Rewriter π_ψ that edits an original scenario $c = (p, d)$ into a rewritten one $\hat{c} = (\hat{p}, \hat{d})$:

$$\hat{c} \sim \pi_\psi(\cdot | c). \quad (4)$$

The Rewriter is prompted to make $\hat{c}$ more complex and more diverse than c , so that it covers scenarios underrepresented in the original pool $\mathcal{S}$ and exposes the Actor’s weaknesses.

To train the Rewriter to expose the weaknesses of the current Actor, we define its reward as the gap between the Actor’s performance on the original and rewritten scenarios. Specifically, we freeze the Actor and sample N responses ($N=3$ in our experiments) for each scenario:

$$\{r^{(i)}\}_{i=1}^N \sim \pi_\phi(\cdot | c), \quad \{\hat{r}^{(i)}\}_{i=1}^N \sim \pi_\phi(\cdot | \hat{c}). \quad (5)$$

The Rewriter’s reward is then computed as:

$$R_{\text{Rewriter}}(\hat{c}) = \frac{1}{N} \sum_{i=1}^N \text{RM}(r^{(i)} | c) - \frac{1}{N} \sum_{i=1}^N \text{RM}(\hat{r}^{(i)} | \hat{c}). \quad (6)$$

Averaging over multiple responses helps reduce the variance from stochastic decoding and reward-model scoring, yielding a more stable estimate of the difficulty gap. A larger $R_{\text{Rewriter}}(\hat{c})$ indicates that the rewritten scenario reveals the weakness for the current Actor, encouraging the Rewriter to discover character profiles and contexts that the Actor has not yet mastered.

We also constrain the length of rewritten scenarios. If $|\hat{c}|$ exceeds 150% of the original length $|c|$, we set $R_{\text{Rewriter}}(\hat{c})$ to -5 , which discourages the Rewriter from padding scenarios with irrelevant content.

We then update the Rewriter by maximizing the GRPO objective in Equation (3), with π_θ instantiated as π_ψ , x as the original scenario c , and R as R_{Rewriter} in Equation (6).

Scenario Augmentation

After updating the Rewriter, we use it to expand the scenario pool. For each original scenario $c \in \mathcal{S}_0$ in the initial pool, the updated Rewriter produces a rewritten version $\hat{c}$, and

Model	Conversational Ability (↑)	Character Consistency (↑)	Role-playing Attractiveness (↑)	Overall (↑)
<i>Large-Scale LLMs</i>				
Kimi-K2.6 (non-thinking)	4.10	3.21	3.43	3.58
Kimi-K2.6 (thinking)	4.06	3.35	3.58	3.66
Claude-3.7-Sonnet	3.90	3.06	3.25	3.41
GPT-4o	3.63	2.90	3.06	3.20
Qwen2.5-72B-Instruct	3.80	3.05	3.28	3.37
<i>Qwen2.5-7B-Instruct</i>				
Vanilla	3.71	2.91	3.13	3.25
CPO (Ye et al. 2025)	3.75	3.03	3.30	3.36
R4 (Wang et al. 2026a)	3.79	2.79	3.05	3.21
VeriRole (Wang et al. 2026b)	3.50	3.16	3.24	3.30
AdvRole (Ours)	3.83	3.10	3.36	3.43
<i>Qwen2.5-14B-Instruct</i>				
Vanilla	3.72	2.95	3.17	3.28
CPO (Ye et al. 2025)	3.81	3.04	3.29	3.38
R4 (Wang et al. 2026a)	3.86	3.00	3.22	3.36
VeriRole (Wang et al. 2026b)	3.56	3.17	3.23	3.32
AdvRole (Ours)	3.89	3.18	3.40	3.49

Table 1: **Performance on the CharacterEval benchmark.** Each dimension score is the arithmetic mean of its underlying metrics between 1 and 5, and “Overall” is the mean of the three dimensions. All baselines re-implemented under identical data splits.

all rewritten scenarios are merged into the pool to form the training data for the next epoch:

$$\mathcal{S} \leftarrow \mathcal{S} \cup \{\hat{c} \mid \hat{c} \sim \pi_{\psi}(\cdot \mid c), c \in \mathcal{S}_0\}. \quad (7)$$

We always rewrite from the original $\mathcal{S}_0$ rather than from previously rewritten scenarios, which keeps each epoch’s augmentation grounded in real character profiles and contexts and avoids drifting too far from the data distribution.

By alternating the three steps across epochs, the scenario pool keeps evolving alongside the Actor, and the Actor is continuously exposed to more diverse scenarios that target its current weaknesses.

Experiments

Implementation Details

We separately train and evaluate AdvRole on two role-playing benchmarks of different languages: CoSER (Wang et al. 2025c) for English and CharacterEval (Tu et al. 2024) for Chinese. To further assess generalizability, we additionally take the model trained on CharacterEval and evaluate it in a zero-shot manner on RAIDEN (Wu et al. 2025), another Chinese role-playing benchmark.

For CoSER, we follow CogDual (Liu et al. 2025) and use 17,762 scenarios as the training set and 200 as the test set. For CharacterEval, we randomly select 2,000 scenarios as the training set and use the remaining 2,564 as the test set. For RAIDEN, we directly evaluate on its test set.

Training Scheme We implement our training pipeline based on the verl framework² (Sheng et al. 2025). For the GRPO hyperparameter settings, we configure the sample

size to $G = 8$, the clipping parameter to $\epsilon = 0.2$, and the KL divergence coefficient to $\beta = 0.04$. The total batch size per step is set to 32. The Actor and the Rewriter are initialized from the same LLM and fine-tuned as two separate models. For the Rewriter reward in Equation (6), we sample $N = 3$ Actor responses per scenario.

The training is conducted with a learning rate of 1×10^{-6} across 16 H20 GPUs, alternating Actor training, Rewriter training, and scenario augmentation for 3 epochs in total.

Evaluation Setup

CoSER. CoSER evaluates an entire multi-turn dialogue rather than a single response. At test time, an auxiliary LLM simulates both the dialogue partner and the environment, conducting a 30-turn conversation with the role-playing model. An LLM judge then rates the dialogue with a penalty-based protocol along four dimensions: Storyline Consistency, Anthropomorphism, Character Fidelity, and Storyline Quality. Each dimension starts from 100 and is deducted according to detected flaws, so all scores fall in the range $[0, 100]$ and higher is better. Following our baselines (Wang et al. 2025c; Liu et al. 2025), we use GPT-4o as both the dialogue partner and the LLM judge.

CharacterEval. CharacterEval evaluates a single response given a character profile and dialogue context, using BaichuanCharRM³ (Tu et al. 2024) to score 12 metrics on a 1–5 scale. The metrics are grouped into three dimensions: Conversational Ability (Fluency, Coherency, Consistency), Character Consistency (covering both knowledge-related and persona-related aspects), and Role-playing Attractiveness (Human-Likeness, Communication Skills, Expression Di-

²<https://github.com/verl-project/verl>

³<https://huggingface.co/morecry/BaichuanCharRM>

Model	Storyline Consistency (↑)	Anthropo- morphism (↑)	Character Fidelity (↑)	Storyline Quality (↑)	Average (↑)	BLEU (↑)	ROUGE-L (↑)
<i>Large-Scale LLMs</i>							
Gemini-1.5-Pro	59.1	52.4	47.8	77.6	59.2	0.054	0.117
DeepSeek-V3	56.4	47.9	44.0	76.7	56.2	0.045	0.110
Claude-3.5-Sonnet	57.5	48.5	45.7	77.2	57.2	0.052	0.115
GPT-4o	58.9	43.1	41.6	75.4	54.8	0.059	0.121
<i>Qwen2.5-7B-Instruct</i>							
Vanilla	59.9	42.0	41.5	62.3	51.4	0.007	0.155
CoT	55.8	37.2	36.5	61.8	47.8	0.007	0.153
CB-CoT	56.9	44.9	39.1	62.5	50.8	0.008	0.155
CogDual-SFT	58.4	47.0	45.0	71.7	55.5	0.015	0.161
CogDual-RL	59.9	46.6	47.0	74.0	56.9	0.016	0.163
AdvRole (Ours)	61.5	48.5	49.2	77.2	59.1	0.015	0.161
<i>LLaMA3.1-8B-Instruct</i>							
Vanilla	48.2	36.6	27.0	63.7	43.9	0.046	0.102
CoT	50.1	40.4	28.0	64.3	45.7	0.047	0.103
CB-CoT	52.8	41.4	27.7	65.0	46.7	0.048	0.105
CogDual-SFT	56.0	46.9	43.8	75.1	55.4	0.061	0.118
CogDual-RL	60.1	45.9	48.8	73.1	57.0	0.064	0.121
AdvRole (Ours)	61.3	47.2	50.3	75.4	58.6	0.060	0.119

Table 2: **Performance on the CoSER benchmark.** The first four columns are LLM-judged dimension scores (0–100). “Average” is the arithmetic mean of the four dimension means. BLEU and ROUGE-L are corpus-level n-gram metrics against the reference dialogue.

versity, Empathy). We report each dimension as the mean of its metrics and an Overall score as the mean of the three dimensions.

RAIDEN. RAIDEN evaluates a single response at predefined dialogue anchor points. We follow the evaluation setting of VeriRole (Wang et al. 2026b), which uses an LLM judge (GPT-4o) to assign a binary correctness score on six fine-grained dimensions and reports the score in [0, 1].

All prompts used in training and evaluation are provided in the supplementary material.

Main Results

We evaluate AdvRole on CharacterEval against three recent role-playing RL methods: CPO (Ye et al. 2025), R4 (Wang et al. 2026a), and VeriRole (Wang et al. 2026b). We use Qwen2.5-7B-Instruct⁴ and Qwen2.5-14B-Instruct⁵ as the base models for fine-tuning.

As shown in Table 1, AdvRole achieves the best overall score under both base models, with consistent improvements across most dimensions. The only exception is character consistency under Qwen2.5-7B-Instruct, on which VeriRole is marginally higher. This is likely because VeriRole explicitly optimizes for character consistency through hint-based rewards, whereas AdvRole distributes its training signal over a broader set of role-playing aspects.

More importantly, our fine-tuned Qwen2.5-14B-Instruct surpasses several large-scale LLMs with far more parameters on the overall score. This suggests that the bottleneck of small open-source role-playing models lies less in raw model capacity than in the limited coverage of their training scenarios, and that adversarially expanding the scenario pool is an effective solution.

We further evaluate AdvRole on CoSER against several baselines from CogDual (Liu et al. 2025). As shown in Table 2, AdvRole achieves the best average score under both base models, with gains on all four LLM-judged dimensions. Against the strongest baseline CogDual-RL, the gain is the largest on storyline quality and the second largest on character fidelity under both base models. This pattern indicates that the rewritten scenarios mainly help the Actor on character profiles and dialogue contexts that the original pool covers sparsely.

On BLEU and ROUGE-L, AdvRole is slightly below CogDual-RL, as our reward targets role-playing quality rather than n-gram overlap with reference dialogues.

Cross-Benchmark Generalization

To examine whether the gains brought by AdvRole transfer beyond the training benchmark, we take the model trained on CharacterEval and directly evaluate it on RAIDEN without any further tuning. As shown in Table 3, AdvRole achieves the best average score. At the dimension level, VeriRole obtains the highest scores on SBK and CM, which are exactly the dimensions its hint-based reward is designed to target through character profiles and dialogue history. In contrast,

⁴<https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>

⁵<https://huggingface.co/Qwen/Qwen2.5-14B-Instruct>

Qwen2.5-7B-Instruct	SBK (↑)	CM (↑)	SCK (↑)	RCB (↑)	TA (↑)	Average (↑)
Vanilla	0.69	0.66	0.61	0.48	0.42	0.57
CPO (Ye et al. 2025)	0.71	0.69	0.63	0.50	0.44	0.59
R4 (Wang et al. 2026a)	0.70	0.68	0.64	0.51	0.43	0.59
VeriRole (Wang et al. 2026b)	0.74	0.72	0.65	0.51	0.44	0.61
AdvRole (Ours)	0.73	0.71	0.68	0.58	0.51	0.64

Table 3: **Zero-shot performance on the RAIDEN benchmark.** All models are trained on CharacterEval and directly evaluated on RAIDEN without further tuning. Following VeriRole (Wang et al. 2026b), each dimension is scored by an LLM judge (GPT-4o). The five dimensions are Script-Based Knowledge (SBK), Conversation Memory (CM), Script-Contradictory Knowledge (SCK), Role-Cognition Boundary (RCB), and Topic Advancement (TA).

Variant	CA (↑)	CC (↑)	RA (↑)	Overall (↑)
Profile only	3.81	3.08	3.33	3.41
Context only	3.79	3.07	3.31	3.39
Profile + Context	3.83	3.10	3.36	3.43
w/o Rewriter	3.74	2.89	3.18	3.27
Frozen Rewriter	3.77	3.05	3.28	3.37
Co-trained Rewriter	3.83	3.10	3.36	3.43

Table 4: **Ablation on the Rewriter.** CA, CC, and RA denote Conversational Ability, Character Consistency, and Role-playing Attractiveness, respectively.

AdvRole leads on SCK, RCB and TA, which require the model to handle out-of-profile situations and steer the conversation in character.

This matches what each method optimizes for. VeriRole’s reward is tied to verifiable hints from profiles and histories, while AdvRole distributes its signal over broader role-playing aspects through rewritten scenarios, leading to stronger behavior beyond profile adherence.

Ablation Study

We conduct ablation studies on CharacterEval with Qwen2.5-7B-Instruct as the base model. To disentangle the contribution of profile and context edits, we compare three variants in Table 4: rewriting only the profile, rewriting only the dialogue context, and rewriting both. The result shows that editing either part alone yields lower scores than editing both. The two rewriting targets stress different weaknesses of the Actor. Edited profiles test how well it adapts to perturbed character settings, while edited contexts test how it responds to unfamiliar dialogue states. Combining them gives the Rewriter more freedom to construct more diverse scenarios and improves the Actor’s performance.

Besides, we compare three Rewriter settings in Table 4. The overall score improves steadily from w/o Rewriter to frozen to co-trained, showing that rewriting itself brings a clear gain and that co-training brings a further one. Without RL updates, the frozen Rewriter cannot fully follow the Actor’s evolving weaknesses, so part of its edits becomes less informative as training goes on. Co-training keeps the rewritten scenarios aligned with the current Actor and makes the adversarial loop between the two roles effective.

Variant	CA (↑)	CC (↑)	RA (↑)	Overall (↑)
$N=1$	3.79	3.07	3.30	3.39
$N=2$	3.82	3.11	3.34	3.42
$N=3$ (Ours)	3.83	3.10	3.36	3.43

Table 5: **Ablation on N .** N is the number of responses sampled per scenario in Equation (5).

To test hyperparameter sensitivity, we vary the number of responses N sampled per scenario when computing the Rewriter’s reward (Equation 6) in Table 5. With $N=1$, the reward relies on a single response for each original and rewritten scenario, making the estimated weakness gap noisy. Increasing N to 2 recovers most of the gain and slightly surpasses $N=3$ on CC, while $N=3$ gives the best overall score. We therefore use $N=3$ as the default, as it provides a stable reward estimate with a moderate rollout cost.

Scenario Pool Diversity

We encode each scenario with text-embedding-3-large and quantify pool diversity with three metrics. Intra-Pool Distance (IPD) is the average pairwise cosine distance among all scenarios in the pool, reflecting overall spread. Embedding Drift (ED) is the average cosine distance between each rewritten scenario and its source, indicating how far the Rewriter pushes a scenario from its origin. New Clusters (NC) is the number of K-Means clusters that contain at least one rewritten scenario but no original scenario, measuring whether the pool expands into previously unoccupied semantic regions.

As shown in Table 6, all three metrics increase steadily across training epochs, indicating that the Rewriter keeps pushing scenarios into unexplored territory as the Actor improves. The supplementary material further analyzes whether these scenarios expose the current Actor’s weaknesses and are later learned by the next Actor.

Lanobe Benchmark

Role-playing agents must handle a broad space of characters, situations, and speaking styles. The core problem is that a fixed training pool covers only a small part of this space, so the model can overfit to seen scenarios. Our experiments on English and Chinese benchmarks show that AdvRole helps by adding character and context pairs that expose model failures during training.

Pool	IPD (↑)	ED (↑)	NC (↑)
S_0 (Original)	0.564	–	–
+ Epoch 1 Rewrite	0.581	0.203	7/30
+ Epoch 2 Rewrite	0.594	0.231	11/30
+ Epoch 3 Rewrite	0.602	0.254	14/30
+ Random Paraphrase	0.566	0.008	0/30

Table 6: **Scenario pool diversity measured in embedding space.** NC is reported out of 30 total K-Means clusters fitted on the original pool. Random Paraphrase simply prompts GPT-4o to rephrase each scenario without any adversarial objective.

This issue is amplified in multilingual settings, where character names, backgrounds, scenes, and dialogue styles are often language-specific. Existing role-playing benchmarks mainly cover English (Wang et al. 2025c) or Chinese (Tu et al. 2024; Wu et al. 2025), so they cannot show whether improving the scenario pool also helps when the language and cultural setting change. To test this, we introduce **Lanobe**, a multilingual role-playing benchmark covering five languages: Simplified Chinese, Traditional Chinese, Japanese, Korean, and Thai. For each language, we provide 1,000 training and 500 test scenarios.

We collect character descriptions and plot summaries from publicly available sources and process them through a multi-stage LLM pipeline powered by Kimi-K2.6.⁶ The full construction pipeline is described in the supplementary material.

For evaluation, we use Kimi-K2.6 as the LLM judge to score each model response along four dimensions on a 0–10 scale: Character Consistency (CC), Situational Coherence (SC), Linguistic Quality (LQ), and Interaction Proactivity (IP). The final score is the arithmetic mean of the four dimensions.

We train all methods on the Lanobe training set using the same setting described in the Implementation Details section. As shown in Table 7, AdvRole achieves the highest average score across all five languages, confirming that the adversarial scenario-rewriting strategy generalizes beyond English-centric benchmarks. Notably, VeriRole obtains the highest CC score on Korean and Thai, consistent with its character-consistency-focused reward design observed in CharacterEval. However, AdvRole still achieves the best overall score in every language by improving more broadly across all four dimensions.

These results also show why multilingual evaluation is needed. AdvRole brings clear gains in Chinese, while the gains on Korean and Thai are smaller. This suggests that improving the scenario pool is not the only factor in multilingual role-playing. The base model’s language ability also matters. When the base model already has stronger language knowledge, RL can better turn targeted training scenarios into role-playing gains. When the target language is weaker

⁶The released benchmark contains only LLM-generated text. No verbatim excerpts from copyrighted material are included, and the benchmark will be released under a non-commercial license for research purposes only.

Qwen2.5-7B-Instruct	CC	SC	LQ	IP	Avg.
<i>Simplified Chinese</i>					
Vanilla	5.99	5.70	7.59	5.04	6.08
CPO	6.19	5.85	7.65	5.29	6.24
R4	6.11	5.65	7.62	4.96	6.09
VeriRole	6.34	5.78	7.54	5.19	6.21
AdvRole (Ours)	6.41	6.05	7.71	5.49	6.42
<i>Traditional Chinese</i>					
Vanilla	5.09	4.98	6.01	4.78	5.21
CPO	5.29	5.13	6.07	5.03	5.38
R4	5.21	4.93	6.04	4.70	5.22
VeriRole	5.44	5.06	5.96	4.93	5.35
AdvRole (Ours)	5.51	5.33	6.13	5.23	5.55
<i>Japanese</i>					
Vanilla	5.32	4.98	6.95	4.52	5.44
CPO	5.52	5.13	7.01	4.77	5.61
R4	5.44	4.93	6.98	4.44	5.45
VeriRole	5.67	5.06	6.90	4.67	5.58
AdvRole (Ours)	5.67	5.26	7.03	4.90	5.71
<i>Korean</i>					
Vanilla	4.70	4.45	6.42	4.00	4.89
CPO	4.90	4.60	6.48	4.25	5.06
R4	4.82	4.40	6.45	3.92	4.90
VeriRole	5.05	4.53	6.37	4.15	5.03
AdvRole (Ours)	5.00	4.70	6.49	4.35	5.13
<i>Thai</i>					
Vanilla	5.49	5.04	6.51	4.97	5.50
CPO	5.69	5.19	6.57	5.22	5.67
R4	5.61	4.99	6.54	4.89	5.51
VeriRole	5.84	5.12	6.46	5.12	5.64
AdvRole (Ours)	5.74	5.24	6.56	5.25	5.70

Table 7: **Performance on the Lanobe benchmark.** CC, SC, LQ, and IP denote Character Consistency, Situational Coherence, Linguistic Quality, and Interaction Proactivity, respectively. All dimensions are scored by Kimi-K2.6 on a 0–10 scale.

in the base model, the same training signal has less room to work.

Conclusion

In this paper, we address the issue that role-playing agents are limited by a fixed pool of training scenarios. We propose AdvRole, an adversarial RL framework in which a Rewriter edits character profiles and dialogue contexts into scenarios that expose the Actor’s weaknesses and is rewarded by the Actor’s performance gap between the original and rewritten ones. Experiments on three role-playing benchmarks, including a new multilingual one we release, show that AdvRole outperforms baselines and offers a new perspective on training role-playing agents by evolving the training data.

References

- Abdulhai, M.; et al. 2026. Consistently Simulating Human Personas with Multi-Turn Reinforcement Learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Agarwal, S.; et al. 2026. The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Chen, J.; et al. 2024. From Persona to Personalization: A Survey on Role-Playing Language Agents. *Transactions on Machine Learning Research*. Survey Certification.
- Chen, Y.; et al. 2023. SoulChat: Improving LLMs' Empathy, Listening, and Comfort Abilities through Fine-tuning with Multi-turn Empathy Conversations. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, 1170–1183. Singapore: Association for Computational Linguistics.
- Cheng, D.; et al. 2026. Reasoning with Exploration: An Entropy Perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(36): 30377–30385.
- Chung, J. J. Y.; et al. 2025. Modifying Large Language Model Post-Training for Diverse Creative Writing. In *Second Conference on Language Modeling*.
- Han, J.-E.; et al. 2024. PSYDIAL: Personality-based Synthetic Dialogue Generation Using Large Language Models. In Calzolari, N.; Kan, M.-Y.; Hoste, V.; Lenci, A.; Sakti, S.; and Xue, N., eds., *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 13321–13331. Torino, Italia: ELRA and ICCL.
- Huang, C.; et al. 2026. R-Zero: Self-Evolving Reasoning LLM from Zero Data. In *The Fourteenth International Conference on Learning Representations*.
- Ji, K.; et al. 2025. Enhancing Persona Consistency for LLMs' Role-Playing using Persona-Aware Contrastive Learning. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Findings of the Association for Computational Linguistics: ACL 2025*, 26221–26238. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-256-5.
- Jiang, H.; et al. 2025. VERSE: verification-based self-play for code instructions. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI'25/IAAI'25/EAAI'25*. AAAI Press. ISBN 978-1-57735-897-8.
- Lan, Y.; et al. 2024. LLM-Based Agent Society Investigation: Collaboration and Confrontation in Avalon Gameplay. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 128–145. Miami, Florida, USA: Association for Computational Linguistics.
- Li, P.; et al. 2025. Confidence Is All You Need: Few-Shot RL Fine-Tuning of Language Models. arXiv:2506.06395.
- Lin, Z.; et al. 2025. Learning to Solve and Verify: A Self-Play Framework for Mutually Improving Code and Test Generation. In *NeurIPS 2025 Fourth Workshop on Deep Learning for Code*.
- Liu, C.; et al. 2025. CogDual: Enhancing Dual Cognition of LLMs via Reinforcement Learning with Implicit Rule-Based Rewards. In Christodoulopoulos, C.; Chakraborty, T.; Rose, C.; and Peng, V., eds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 27307–27336. Suzhou, China: Association for Computational Linguistics. ISBN 979-8-89176-332-6.
- Lu, K.; et al. 2024. Large Language Models are Superpositions of All Characters: Attaining Arbitrary Role-play via Self-Alignment. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 7828–7840. Bangkok, Thailand: Association for Computational Linguistics.
- Ouyang, L.; et al. 2022. Training language models to follow instructions with human feedback. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.
- Peng, L.; and Shang, J. 2024. Quantifying and Optimizing Global Faithfulness in Persona-driven Role-playing. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Prabhudesai, M.; et al. 2025. Maximizing Confidence Alone Improves Reasoning. arXiv:2505.22660.
- Prasad, A.; et al. 2025. Self-Consistency Preference Optimization. In *Forty-second International Conference on Machine Learning*.
- Rafailov, R.; et al. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Rashkin, H.; et al. 2025. Help Me Write a Story: Evaluating LLMs' Ability to Generate Writing Feedback. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 25827–25847. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.
- Saakyan, A.; et al. 2026. Death of the Novel(ty): Beyond N-Gram Novelty as a Metric for Textual Creativity. In *The Fourteenth International Conference on Learning Representations*.
- Shao, Y.; et al. 2023. Character-LLM: A Trainable Agent for Role-Playing. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 13153–13187. Singapore: Association for Computational Linguistics.
- Shao, Z.; et al. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *CoRR*, abs/2402.03300.
- Sheng, G.; et al. 2025. HybridFlow: A Flexible and Efficient RLHF Framework. In *Proceedings of the Twentieth*

- European Conference on Computer Systems*, EuroSys '25, 1279–1297. New York, NY, USA: Association for Computing Machinery. ISBN 9798400711961.
- Tu, Q.; et al. 2024. CharacterEval: A Chinese Benchmark for Role-Playing Conversational Agent Evaluation. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 11836–11850. Bangkok, Thailand: Association for Computational Linguistics.
- Wang, F.; et al. 2025a. Flexible Thinking for Multimodal Emotional Support Conversation via Reinforcement Learning. In Christodoulopoulos, C.; Chakraborty, T.; Rose, C.; and Peng, V., eds., *Findings of the Association for Computational Linguistics: EMNLP 2025*, 1341–1356. Suzhou, China: Association for Computational Linguistics. ISBN 979-8-89176-335-7.
- Wang, N.; et al. 2024a. RoleLLM: Benchmarking, Eliciting, and Enhancing Role-Playing Abilities of Large Language Models. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 14743–14777. Bangkok, Thailand: Association for Computational Linguistics.
- Wang, R.; et al. 2026a. R4: Nested Reasoning-Retrieval for Reward Modeling in Role-Playing Agents. In *The Fourteenth International Conference on Learning Representations*.
- Wang, X.; et al. 2024b. InCharacter: Evaluating Personality Fidelity in Role-Playing Agents through Psychological Interviews. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1840–1873. Bangkok, Thailand: Association for Computational Linguistics.
- Wang, X.; et al. 2025b. CoSER: Coordinating LLM-Based Persona Simulation of Established Roles. In *Forty-second International Conference on Machine Learning*.
- Wang, X.; et al. 2025c. CoSER: Coordinating LLM-Based Persona Simulation of Established Roles. In *Forty-second International Conference on Machine Learning*.
- Wang, Z.; et al. 2025d. RAIDEN-R1: Improving Role-awareness of LLMs via GRPO with Verifiable Reward. arXiv:2505.10218.
- Wang, Z.; et al. 2026b. VeriRole: Verifiable Role-Awareness through Hint-Guided Reinforcement Learning. In *The Fourteenth International Conference on Learning Representations*.
- Wei, X.; et al. 2025. Igniting Creative Writing in Small Language Models: LLM-as-a-Judge versus Multi-Agent Refined Rewards. In Christodoulopoulos, C.; Chakraborty, T.; Rose, C.; and Peng, V., eds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 17160–17186. Suzhou, China: Association for Computational Linguistics. ISBN 979-8-89176-332-6.
- Wu, B.; et al. 2025. RAIDEN Benchmark: Evaluating Role-playing Conversational Agents with Measurement-Driven Custom Dialogues. In Rambow, O.; Wanner, L.; Apidianaki, M.; Al-Khalifa, H.; Eugenio, B. D.; and Schockaert, S., eds., *Proceedings of the 31st International Conference on Computational Linguistics*, 11086–11106. Abu Dhabi, UAE: Association for Computational Linguistics.
- Ye, J.; et al. 2025. CPO: Addressing Reward Ambiguity in Role-playing Dialogue via Comparative Policy Optimization. In Christodoulopoulos, C.; Chakraborty, T.; Rose, C.; and Peng, V., eds., *Findings of the Association for Computational Linguistics: EMNLP 2025*, 297–323. Suzhou, China: Association for Computational Linguistics. ISBN 979-8-89176-335-7.
- Yuan, J.; et al. 2025. ReflectDiffu: Reflect between Emotion-intent Contagion and Mimicry for Empathetic Response Generation via a RL-Diffusion Framework. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 25435–25449. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.
- Zeng, Z.; et al. 2025. PersLLM: A Personified Training Approach for Large Language Models. In Christodoulopoulos, C.; Chakraborty, T.; Rose, C.; and Peng, V., eds., *Findings of the Association for Computational Linguistics: EMNLP 2025*, 8598–8617. Suzhou, China: Association for Computational Linguistics. ISBN 979-8-89176-335-7.
- Zhang, K.; et al. 2026. Consistent Paths Lead to Truth: Self-Rewarding Reinforcement Learning for LLM Reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Zhang, Z.; et al. 2025a. DVM: Towards Controllable LLM Agents in Social Deduction Games. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5.
- Zhang, Z.; et al. 2025b. MultiMind: Enhancing Werewolf Agents with Multimodal Reasoning and Theory of Mind. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, 5824–5833. New York, NY, USA: Association for Computing Machinery. ISBN 9798400720352.
- Zhao, A.; et al. 2026. Absolute Zero: Reinforced Self-play Reasoning with Zero Data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Zheng, X.; et al. 2025. Customizing Emotional Support: How Do Individuals Construct and Interact With LLM-Powered Chatbots. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25. New York, NY, USA: Association for Computing Machinery. ISBN 9798400713941.
- Zhou, J.; et al. 2024. CharacterGLM: Customizing Social Characters with Large Language Models. In Dernoncourt, F.; PreoŃuc-Pietro, D.; and Shimorina, A., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 1457–1476. Miami, Florida, US: Association for Computational Linguistics.
- Zhou, Y.; et al. 2026. Self-Challenging Language Model Agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Zuo, Y.; et al. 2026. TTRL: Test-Time Reinforcement Learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.