

# From Homogeneity to Diversity: Self-Refined GRPO for Controllable Creative Writing

Minna Peng<sup>1,2</sup>, Wei Tan<sup>1\*</sup>, Zeyu Wang<sup>1,3</sup>, Yongquan Hu<sup>1</sup>, Ruizhi Huang<sup>1</sup>, Jie Yang<sup>1</sup>

<sup>1</sup>International Digital Economy Academy (IDEA)

<sup>2</sup>Guangdong University of Foreign Studies

<sup>3</sup>The Hong Kong University of Science and Technology (Guangzhou)  
pengminna@mail.gdufs.edu.cn, tanwei@idea.edu.cn,  
zwang312@connect.hkust-gz.edu.cn

## Abstract

Controllable creative writing is a text generation task that requires both high generation quality and diverse outputs. However, existing methods often improve one aspect at the expense of the other. To address this challenge, we propose SERGE, a reinforcement learning (RL) framework that introduces a diversity-guided self-refinement mechanism, where the policy itself refines low-diversity candidate texts, and employs policy shaping to amplify the gradient contributions of samples with both high generation quality and diversity. We also propose a fine-grained reward modeling framework to provide reliable quality reward signals to support RL optimization. Experiments on the short-form video copywriting generation task show that SERGE outperforms the base model and conventional diversity-enhancement strategies on both diversity metrics (DisNGram, PairBLEU, CosSim) and generation quality, effectively alleviating the “diversity collapse” issue in RL. Our code is available at: <https://github.com/peng-mn/SERGE>.

## 1 Introduction

Controllable creative writing is a crucial task in modern AI-driven content creation, aiming to automatically generate stylistically engaging text for a given topic while adhering to predefined constraints such as length requirements or structural guidelines. Unlike factual generation tasks such as mathematical reasoning or code generation, which often target a single correct solution (Cobbe et al., 2021; Chen, 2021), controllable creative writing typically admits numerous semantically valid but stylistically diverse outputs for the same input (Deshpande et al., 2025; Fan et al., 2018), catering to varied user preferences and application scenarios (Wang et al., 2025). Existing work on controllable creative writing commonly focuses on two direc-

tions: improving generation diversity or enhancing generation quality.

To increase diversity, previous studies have proposed methods such as adjusting the temperature (Chung et al., 2023), adopting different decoding strategies (Fan et al., 2018; Holtzman et al., 2019; Nguyen et al., 2024), optimizing the prompt design (Hayati et al., 2024; Zhang et al., 2025), or modifying the maximum likelihood training objective (Li et al., 2016a). However, previous studies have shown that these approaches often improve diversity at the cost of generation quality (Zhang et al., 2025; Padmakumar et al., 2025).

For quality optimization, RL has achieved success in both tasks with verifiable answers (Shao et al., 2024) and controllable creative writing (Wei et al., 2025). However, RL often induces output homogeneity, commonly known as “diversity collapse”, where outputs become overly concentrated on high-reward trajectories, producing increasingly similar or even identical results. (Kirk et al., 2023; Yang and Holtzman, 2025; Huang et al., 2025; Li et al., 2024; O’Mahony et al., 2024; Ji et al., 2025). Entropy regularization can partially alleviate this, but quality degradation remains a concern (Slocum et al., 2025; Lanchantin et al., 2025b). Methods such as Differential Smoothing (Gai et al., 2025) rely on a single correct answer, limiting their applicability to controllable creative writing.

Despite progress in improving either diversity or quality, existing methods often trade one for the other. Thus, a core challenge in controllable creative writing remains: **how to achieve stable and controllable diversity within an RL framework without compromising generation quality.**

To address this challenge, we propose SERGE (SElf-Refined GRPO for controllable crEative writing). To tackle the lack of explicit supervision and the high subjectivity and noise in quality evaluation, we introduce a multi-dimensional, hierarchically weighted evaluation mechanism to construct prefer-

\*Corresponding author

ence pairs with high consistency and clear discriminative power. These pairs are used to train a reward model that captures relative generation quality, providing a stable and reliable optimization signal for subsequent RL.

Building on this, we further propose a novel diversity-guided self-refined GRPO algorithm, an online RL approach designed to address the high similarity among multiple responses sampled from the same policy for a single topic. During training, low-diversity samples trigger a self-refinement mechanism that produces candidates with greater expressive variation. This expands the coverage of sampled responses, providing more diverse and informative learning signals for policy updates and enabling the joint optimization of generation quality and diversity. Since newly generated diverse patterns often have low probabilities and are under-represented in policy updates, we employ policy shaping to amplify the gradient contributions of high-quality, high-diversity self-refinement behaviors.

Additionally, to support reward model training and RL optimization in a practical controllable creative writing scenario, we construct a real-world dataset of short-form video copywriting collected from short-form video platforms

Our contributions can be summarized as follows:

- We propose a fine-grained reward modeling approach with hierarchically weighted multi-dimensional evaluation to improve reward stability and discrimination.
- We propose a novel diversity-guided self-refined GRPO method, in which self-refinement by the policy model is introduced to jointly optimize generation quality and diversity.
- To facilitate reward model training and RL optimization, we construct a real-world short-form video copywriting dataset collected from short-form video platforms.
- We conduct systematic experiments demonstrating that SERGE effectively alleviates “diversity collapse” while maintaining or even enhancing generation quality in controllable creative writing.

## 2 Related Work

Large language models often produce homogenized outputs, making insufficient diversity a criti-

cal issue (Jiang et al., 2025). Prior work addresses this mainly at two stages: inference and training.

Decoding-based strategies during inference increase output variability, e.g., via temperature scaling (Tevet and Berant, 2021; Chung et al., 2023), top-k/p sampling (Fan et al., 2018; Holtzman et al., 2019), or iterative and role-conditioned generation (Hayati et al., 2024; Zhang et al., 2025). While effective in boosting diversity, these approaches often reduce generation quality and consistency (Zhang et al., 2025; Padmakumar et al., 2025).

Training-based methods encourage diversity by modifying the objective, e.g., via mutual information maximization (Li et al., 2016a), or repetition penalties (Welleck et al., 2019). More recent work incorporates diversity into preference data for DPO training (Lanchantin et al., 2025a), with the aim of jointly optimizing quality and diversity. However, these are largely applied offline, limiting their effect on online policy updates.

RL-based post-training has been shown to improve the likelihood of producing correct outputs (Shao et al., 2024), but often leads to severe “diversity collapse” (Kirk et al., 2023; Yang and Holtzman, 2025), where the outputs of the model become increasingly concentrated and repetitive. This collapse occurs because policy optimization gradually concentrates probability mass on a few high-reward trajectories, sharpening the output distribution (Huang et al., 2025; Li et al., 2024) and reducing overall diversity (Padmakumar and He, 2023). In extreme cases, the model may generate highly similar or nearly identical outputs across different inputs or multiple samplings, including repeated prefixes (Ji et al., 2025) or almost exact duplication (O’Mahony et al., 2024). Even with entropy regularization (Slocum et al., 2025; Lanchantin et al., 2025b) or global reward reweighting (He et al., 2025a), it remains challenging to maintain both quality and diversity, and training stability can be affected. Gai et al. (2025) analyze this phenomenon theoretically and achieve finer entropy control by distinguishing correct and incorrect trajectories in reasoning tasks. However, such methods rely on well-defined correctness, which is unavailable in controllable creative writing task.

In contrast, we focus on controllable creative writing and address “diversity collapse” directly in RL. Our model, SERGE, jointly models generation quality and diversity, guiding policy updates to improve both simultaneously without relying on auxiliary models or task-specific heuristics.

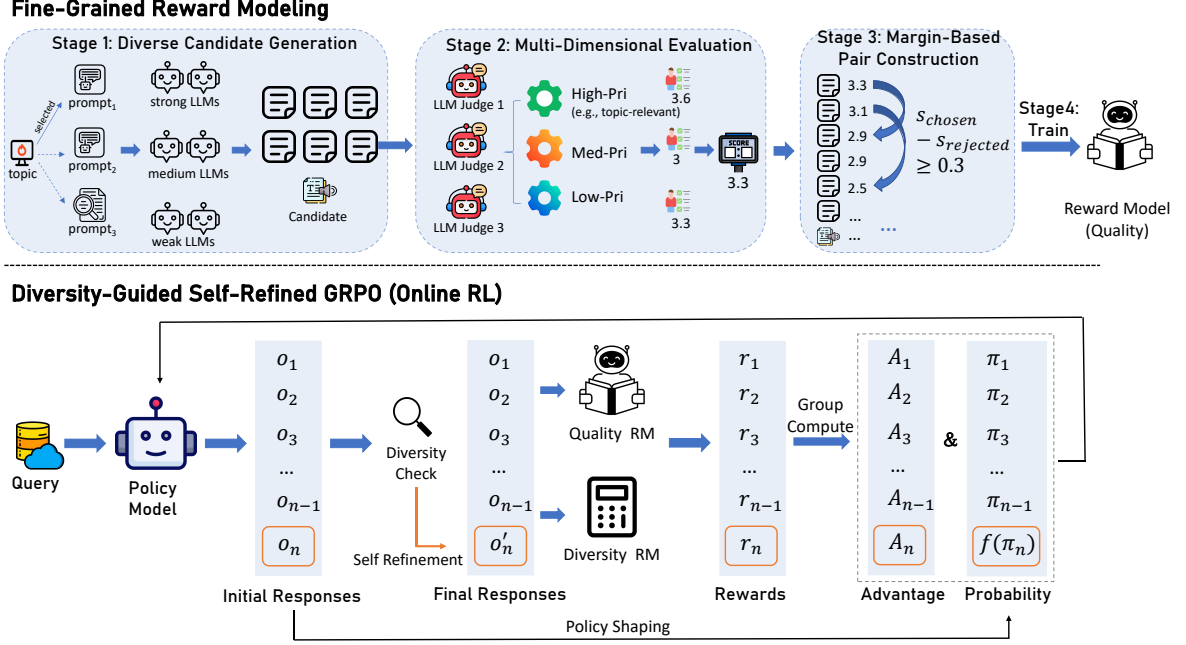


Figure 1: Overview of the SERGE framework. Fine-grained reward modeling is first performed to train a reward model for assessing text quality, followed by self-refined GRPO with diversity-guided refinement for online policy optimization.

### 3 Method

To address “diversity collapse” in RL, we propose **Self-Refined GRPO** for controllable crEative writing (SERGE). As shown in Figure 1, SERGE has two main components. First, a fine-grained reward modeling module constructs high-confidence preference pairs via multi-stage candidate generation and multi-dimensional evaluation, training a reward model that provides reliable quality signals for RL. Second, during online policy optimization, SERGE employs a diversity-guided self-refined GRPO algorithm to mitigate diversity degradation that can occur when improving generation quality. All prompt templates are provided in the Appendix D.

#### 3.1 Fine-Grained Reward Modeling

In this work, generic scalar reward models are not directly adopted because they often perform poorly on specific controllable creative writing tasks (see results in Section 5.2). We also explored an LLM-as-a-judge approach, but found that the evaluation model  $\mathcal{M}_{\text{eval}}$  achieved only about 60% agreement on the final evaluation set (Section 4.1), while incurring substantial training cost. Therefore, we instead design a fine-grained reward modeling framework tailored for controllable creative writing (Wang and

Sun, 2025).

**Stage 1: Diverse Candidate Generation.** This stage constructs a diverse candidate set for each topic, covering different writing styles, expression strategies, and content presentations.

For each topic, a prompt  $p_{\text{selected}}$  is randomly chosen from  $\mathcal{P}$ , and each model  $M_l \in \mathcal{M}_{\text{gen}}$  generates a candidate  $c_{ij}^{\text{gen}} = M_l(q_i, p_{\text{selected}}, L_i)$ . The generated candidates form the set  $C_i^{\text{gen}} = \{c_{ij}^{\text{gen}} \mid M_l \in \mathcal{M}_{\text{gen}}\}$ , where  $q_i \in \mathcal{T}$  is the  $i$ -th topic,  $L_i$  is the target length,  $\mathcal{P} = \{p_1, p_2, p_3\}$  are prompt templates,  $\mathcal{M}_{\text{gen}} = \{M_1, \dots, M_6\}$  are generation models of varying scales, and  $\mathcal{X}$  provides additional context for template  $p_3$  via retrieval. To further diversify, we include manually authored or transcribed text, corrected by an LLM:  $c_i^{\text{human}} = M_{\text{correct}}(q_i, p_{\text{correct}}, c_{\text{human}})$ .

Thus, the final candidate set is defined as  $C_i = C_i^{\text{gen}} \cup \{c_i^{\text{human}}\}$ . This set combines model-generated and human-written text, providing content and stylistic diversity for the subsequent evaluation stage.

**Stage 2: Multi-Dimensional Evaluation.** To obtain fine-grained quality signals for text, each candidate is mapped to a multi-dimensional evaluation space. Let the evaluation space consist of

$d$  independent dimensions (e.g., relevance, information completeness). For a candidate  $c_{ij} \in C_i$  under topic  $q_i$ , its evaluation on the  $v$ -th dimension is defined as  $e_{ij}^{(v)} = E_{\text{LLM}}(c_{ij}, q_i, v)$ , where  $e_{ij}^{(v)} \in \{\text{“Good”}, \text{“Medium”}, \text{“Bad”}\}$  and  $E_{\text{LLM}}(\cdot)$  denotes the LLM-based evaluation function. To ensure high-confidence supervision, only the “Good” label contributes positively to the final score, while “Medium” and “Bad” are treated as zero.

To map multi-dimensional discrete evaluations to a continuous quality score, we introduce a dimension weight vector  $\mathbf{w} = \{w_1, \dots, w_d\}$ . The base quality score of candidate  $c_{ij}$  under a single evaluation model is computed as:

$$s_{ij}^{(m)} = \sum_{v=1}^d w_v \cdot \mathbb{I}[e_{ij}^{(v,m)} = \text{“Good”}] + w_{\text{len}} \cdot f_{\text{len}}(c_{ij}), \quad (1)$$

where  $f_{\text{len}}(\cdot)$  is a length constraint function,  $w_{\text{len}}$  is its associated weight, and  $\mathbb{I}[\cdot]$  is the indicator function.

To mitigate bias from a single evaluation model, we adopt a multi-model ensemble strategy. The final score of candidate  $c_{ij}$  is the mean over all evaluation models:

$$\bar{s}_{ij} = \frac{1}{|\mathcal{M}_{\text{eval}}|} \sum_{M_m \in \mathcal{M}_{\text{eval}}} s_{ij}^{(m)}. \quad (2)$$

After this stage, the aggregated scores of all candidates under topic  $q_i$  are  $S_i = \{\bar{s}_{ij} \mid c_{ij} \in C_i\}$ , which serve as supervision signals for the subsequent ranking and policy optimization stages.

**Stage 3: Margin-Based Pair Construction.** To avoid ambiguous preference signals, we construct preference pairs using a margin constraint after obtaining the aggregated scores of candidate texts (He et al., 2025b). For topic  $q_i$ , let  $C_i$  denote its candidate set and  $S_i$  the corresponding aggregated scores, sorted in descending order. The preference pair construction rule is  $\mathcal{P}_i = \{(c_{i(j)}, c_{i(k)}) \mid j < k, \bar{s}_{i(j)} - \bar{s}_{i(k)} \geq \delta\}$ , where  $c_{i(j)}$  denotes the  $j$ -th candidate after sorting. This construction effectively treats higher-ranked text as *chosen*, while lower-ranked text satisfying the minimum score difference  $\delta$  is treated as *rejected*. The resulting preference dataset is  $D_{\text{pref}} = \bigcup_i \mathcal{P}_i$ .

**Stage 4: Reward Model Training.** The constructed preference dataset  $D_{\text{pref}}$  is used to train the reward model  $\pi_{\text{quality}}$ , with model parameters optimized using the standard pairwise preference learning loss (Hu et al., 2024).

### 3.2 Diversity-Guided Self-Refined GRPO

#### Step 1: Initial Rollout and Diversity Estimation.

Given a topic  $q$  and a target generation length  $L$ , we first sample  $n$  candidate outputs from the old policy  $\pi_{\text{old}}$  for each topic using the initial generation prompt  $I_{\text{init}}$ , such that  $\mathcal{O}_{\text{init}} = \{o_{\text{init}}^{(i)}\}_{i=1}^n$  with  $o_{\text{init}}^{(i)} \sim \pi_{\text{old}}(\cdot \mid I_{\text{init}}, q, L)$  for each  $i$ .

Next, for each generated output  $o_{\text{init}}^{(i)}$ , we compute an initial diversity reward  $R_{i,\text{init}}^{\text{diversity}}$ , which measures the semantic difference of this sample relative to other samples in the current candidate set. Specifically, we define the sample-level diversity reward based on  $N$ -gram coverage (Li et al., 2016b). For a sample  $o_{\text{init}}^{(i)}$ , the diversity reward is defined as:

$$R_{i,\text{init}}^{(i)} = \frac{|\mathcal{G}_N(o_{\text{init}}^{(i)}) \setminus \bigcup_{j \neq i} \mathcal{G}_N(o_{\text{init}}^{(j)})|}{|\mathcal{G}_N(o_{\text{init}}^{(i)})|}, \quad (3)$$

where  $\mathcal{G}_N(\cdot)$  denotes the set of  $N$ -grams extracted from the text.

#### Step 2: Diversity-Guided Self-Refinement.

For generated outputs with low diversity rewards in the initial candidate set, we introduce a self-refinement mechanism to enhance diversity. Specifically, for an initial output  $o_{\text{init}}^{(i)}$ , the policy model is prompted with the self-refinement instruction  $I_{\text{refine}}$  to produce a corresponding refinement:  $o_{\text{ref}}^{(i)} \sim \pi_{\text{old}}(\cdot \mid I_{\text{refine}}, q, L, o_{\text{init}}^{(i)})$ . This yields a set of self-refined candidates  $\mathcal{O}_{\text{ref}} = \{o_{\text{ref}}^{(i)}\}_{i=1}^n$ . Next, initial candidates  $\mathcal{O}_{\text{init}}$  are ranked according to the initial diversity rewards  $R_{i,\text{init}}^{\text{diversity}}$ , and the  $z$  samples with the lowest diversity are identified as  $\mathcal{O}_{\text{low-div}}$ . For each sample in  $\mathcal{O}_{\text{low-div}}$ , its corresponding self-refined output is selected to form a replacement set  $\mathcal{O}_{\text{ref-sel}} \subset \mathcal{O}_{\text{ref}}$ , where  $|\mathcal{O}_{\text{ref-sel}}| = z$ . The remaining samples in the initial candidates are denoted as  $\mathcal{O}_{\text{orig}} = \mathcal{O}_{\text{init}} \setminus \mathcal{O}_{\text{low-div}}$ .

By performing this replacement, we construct a diversity-enhanced candidate set:  $\mathcal{O}_{\text{final}} = \mathcal{O}_{\text{orig}} \cup \mathcal{O}_{\text{ref-sel}}$ . This procedure enables the model to actively explore new expression patterns using its own generated outputs without introducing additional external supervision.

#### Step 3: Reward Design.

After obtaining the refined set of generated text candidates  $\mathcal{O}_{\text{final}}$ , we compute the reward for each candidate  $o_{\text{final}}^{(i)} \in \mathcal{O}_{\text{final}}$ . Specifically, the quality reward  $R_{\text{quality}}^{(i)}$  is computed using the previously trained

preference-based reward model  $\pi_{\text{quality}}$ :  $R_{\text{quality}}^{(i)} = \pi_{\text{quality}}(o_{\text{final}}^{(i)} \mid q, L)$ , which reflects the relative generation quality of each text. The diversity reward  $R_{\text{diversity}}^{(i)}$  is defined as:

$$R_{\text{diversity}}^{(i)} = \frac{|\mathcal{G}_N(o_{\text{final}}^{(i)}) \setminus \bigcup_{j \neq i} \mathcal{G}_N(o_{\text{final}}^{(j)})|}{|\mathcal{G}_N(o_{\text{final}}^{(i)})|}. \quad (4)$$

The final reward for RL is defined as:

$$R^{(i)} = R_{\text{quality}}^{(i)} + \lambda \cdot R_{\text{diversity}}^{(i)}, \quad (5)$$

where  $\lambda$  is a balancing coefficient controlling the weight of the diversity signal.

**Step 4: Online Policy Optimization.** Based on training rewards  $\{R^{(i)}\}$ , we update the online policy using GRPO (Shao et al., 2024). For the diversity-enhanced candidate set  $\mathcal{O}_{\text{final}}$  generated for each topic, GRPO estimates the token-level advantage based on the relative rewards within the group, avoiding explicit modeling of the value function. The token-level advantage for each text  $o_{\text{final}}^{(i)}$  is defined as:

$$A_t^{(i)} = \frac{R^{(i)} - \mu_R}{\sigma_R}, \quad t = 1, \dots, |o_{\text{final}}^{(i)}|, \quad (6)$$

where  $\mu_R$  and  $\sigma_R$  denote the mean and standard deviation of the rewards in  $\mathcal{O}_{\text{final}}$ , respectively.

The training objective of self-refined GRPO is defined as:

$$\begin{aligned} J(\theta) = & \mathbb{E}_{t \sim \mathcal{T}, \mathcal{O}_{\text{orig}}, \mathcal{O}_{\text{ref-sel}}} \left[ \frac{1}{|\mathcal{O}_{\text{orig}}|} \sum_{i \in \mathcal{O}_{\text{orig}}} \sum_{t=1}^{|o_{\text{orig}}^{(i)}|} r_t^{(i)}(\theta) \cdot A_t^{(i)} \right. \\ & + \frac{1}{|\mathcal{O}_{\text{ref-sel}}|} \sum_{i \in \mathcal{O}_{\text{ref-sel}}} \sum_{t=1}^{|o_{\text{ref-sel}}^{(i)}|} f(r_{\text{ref-sel},t}^{(i)}(\theta)) \cdot A_t^{(i)} \\ & \left. - \beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right]. \end{aligned} \quad (7)$$

The token-level probability ratio of the current policy  $\pi_{\theta}$  is

$$r_t^{(i)}(\theta) = \frac{\pi_{\theta}(o_{\text{orig},t}^{(i)} \mid q, o_{\text{orig},<t}^{(i)}, L)}{\pi_{\text{old}}(o_{\text{orig},t}^{(i)} \mid q, o_{\text{orig},<t}^{(i)}, L)}, \quad t = 1, \dots, |o_{\text{orig}}^{(i)}|. \quad (8)$$

Since self-refined candidates are generated under refinement instructions rather than the original sampling distribution, the resulting optimization

is formally off-policy. To account for this mismatch, we adopt a policy shaping strategy with importance weighting following Yan et al. (Yan et al., 2025), thereby prioritizing learning from high-quality and high-diversity outputs while suppressing low-advantage patterns. Formally, for each token in the self-refined candidate  $o_{\text{ref-sel}}^{(i)}$ , the plasticity function is:

$$f(r_{\text{ref-sel},t}^{(i)}(\theta)) = \frac{\pi_{\theta}(o_{\text{ref-sel},t}^{(i)} \mid q, o_{\text{ref-sel},<t}^{(i)}, L)}{\pi_{\theta}(o_{\text{ref-sel},t}^{(i)} \mid q, o_{\text{ref-sel},<t}^{(i)}, L) + \gamma}. \quad (9)$$

This function reweights token-level gradients so that self-refined candidates with larger advantages contribute more strongly to policy updates, enabling the model to jointly improve generation quality and enhance diversity within each candidate group.

## 4 Experimental Setup

### 4.1 Datasets and Metrics

In this work, we conduct experiments on short-form video copywriting generation to evaluate the effectiveness of the proposed approach.

**Reward Model Training Data.** To train a preference-based reward model, we collected short-form video topics and their corresponding video content from 2024–2025 and used ASR to transcribe the audio into text. To align with practical short-form video copywriting scenarios, we retained only copywriting with lengths between 100 and 1000 characters. The raw topics, mostly crawled from the web, contained noisy expressions and varying quality. To improve topic quality, we selected high-performing video copywriting as input and generated topic names using a large model  $M_{\text{topic}}$ , followed by manual verification to replace low-quality topics. This process resulted in 899 valid topics for constructing the reward model training data. Each topic was paired with 6 large model-generated copywriting outputs and 1 original ASR transcript, totaling 6,642 copywriting instances, which produced 5,286 preference pairs for reward model training. All data were anonymized and filtered to remove personally identifying or offensive content. Details on the specific large language models (LLMs) used in data construction are provided in Appendix A.

**Policy Optimization (GRPO) Data.** For the RL stage, we followed the same topic name generation

process introduced above, resulting in 1,815 topics for training.

**Final Evaluation Set.** We randomly selected 100 topics to form the final evaluation set. Each topic was associated with 6 large model-generated copywriting outputs and 1 original human-written transcript, yielding 7 instances per topic. During evaluation, 2 copywriting outputs were randomly sampled from each topic to form pairwise *chosen-rejected* pairs. These pairs were used to evaluate generation quality, diversity, and model-human agreement (details provided in Appendix B).

**Metrics.** We primarily use human annotation agreement to quantify both the evaluation model  $\mathcal{M}_{\text{eval}}$  credibility and reward model scoring accuracy. Additionally, the performance of the RL policy-generated copywriting is assessed along two dimensions: generation quality and diversity. Quality is measured using the reward model trained in Section 3.1, while diversity is quantified using DisNGram (Li et al., 2016b), PairBLEU (Papineni et al., 2002), and cosine similarity (CosSim) (Chowdhury, 2010) to capture lexical and semantic variations among copywriting outputs under the same topic.

## 4.2 Compared Methods

In our experiments, we use Qwen2.5-7B-Instruct and Qwen3-8B-Instruct (Yang et al., 2025) (non-thinking mode) as base models, performing direct inference as reference baselines without RL. Based on these, we implement and compare several inference-time diversity enhancement methods as well as training-based optimization methods.

- Skywork-Reward-V2 (Quality Only): Built upon the base model, the Skywork-Reward-V2-Llama-3.1-8B reward model is used to examine whether off-the-shelf reward signals can effectively support diverse copywriting generation, which serves as the GRPO optimization signal (Liu et al., 2025a). This reward model is adopted because it achieves top performance on the RewardBench leaderboard<sup>1</sup>.
- $\pi_{\text{quality}}$  (Quality Only): The reward model trained in Section 3.1 is used to score generated copywriting for quality, and GRPO is applied based on these scores.

- Inference-time Diversity Enhancement Methods:

- $\pi_{\text{quality}}$  ( $T=1.5/T=2$ ) (Chung et al., 2023): Diversity is enhanced during decoding by increasing the sampling temperature while keeping the same trained policy.
- $\pi_{\text{quality}}$  + Prompt Engineering (Zhang et al., 2025): Diversity-oriented prompts are additionally introduced during inference to explicitly encourage stylistic variation in generated copywriting.

- Training-based Diversity Enhancement Methods:

- $\pi_{\text{quality}}$  + Entropy Regularization (Slocum et al., 2025):  $\pi_{\text{quality}}$  is extended with an entropy regularization term to encourage policy exploration, thus improving generation diversity (Schulman et al., 2017).
- Quality-Diversity Reward (He et al., 2025a): The linear combination of quality and diversity scores in (5) is optimized without introducing any self-refinement mechanism (Wang et al., 2025).

- SERGE (w/o policy shaping): The proposed method is applied without the policy shaping component.

## 4.3 Implementation Details

To more directly and stably capture lexical-level diversity, we set the N-gram order to 1 and use the jieba tokenizer for all diversity score calculations in (3) and (4), as well as related evaluations. This setting mitigates sparsity issues from higher-order N-grams and more reliably reflects lexical differences across generated outputs. For multi-dimensional evaluation,  $d = 14$  with weights 0.3, 0.2, 0.1 for high, medium, low priorities, and a length constraint weight  $w_{\text{len}} = 0.2$ .

During RL training, the target length of each topic is randomly sampled from {280, 450, 700} to ensure coverage of different content lengths. We perform a total of 168 training steps, sampling 8 candidate responses per prompt, with both sampling temperature and top- $p$  set to 1. To maintain consistency between training and inference, the same temperature and top- $p$  configurations are used during inference after training.

For the  $\pi_{\text{quality}}$  + Entropy Regularization method, the entropy regularization coefficient is set to 0.005, balancing stable generation quality with moderate

<sup>1</sup><https://huggingface.co/spaces/allenai/reward-bench>

| Method                                                                  | Qwen2.5-7b-Instruct |                       |                     |                | Qwen3-8b-Instruct   |                       |                     |                |
|-------------------------------------------------------------------------|---------------------|-----------------------|---------------------|----------------|---------------------|-----------------------|---------------------|----------------|
|                                                                         | DisNGram $\uparrow$ | PairBLEU $\downarrow$ | CosSim $\downarrow$ | RMS $\uparrow$ | DisNGram $\uparrow$ | PairBLEU $\downarrow$ | CosSim $\downarrow$ | RMS $\uparrow$ |
| Base                                                                    | 0.2527              | 0.4599                | 0.8736              | -9.71          | 0.2143              | 0.5289                | 0.9066              | -2.08          |
| +Skywork-Reward-V2 (Quality Only)                                       | 0.2635              | 0.4787                | 0.8843              | 0.30           | 0.2167              | 0.5367                | 0.9070              | 9.21           |
| + $\pi_{\text{quality}}$ (Quality Only)                                 | 0.3677              | 0.4545                | 0.8717              | 38.82          | 0.3002              | 0.4994                | 0.8999              | 37.77          |
| + $\pi_{\text{quality}}$ ( $T=1.5$ ) (Chung et al., 2023)               | 0.4084              | 0.4206                | 0.8664              | 39.13          | 0.3527              | 0.4599                | 0.8911              | 37.99          |
| + $\pi_{\text{quality}}$ ( $T=2$ ) (Chung et al., 2023)                 | 0.4320              | <b>0.3931</b>         | 0.8659              | 38.60          | 0.3899              | 0.4292                | 0.8829              | 37.34          |
| + $\pi_{\text{quality}}$ + Prompt Engineering (Zhang et al., 2025)      | 0.4210              | 0.4234                | 0.8446              | 37.88          | 0.3653              | 0.4751                | 0.8896              | 37.74          |
| + $\pi_{\text{quality}}$ + Entropy Regularization (Slocum et al., 2025) | 0.3576              | 0.4229                | 0.8682              | 35.10          | 0.3253              | 0.4676                | 0.8880              | 37.29          |
| + Quality-Diversity Reward (He et al., 2025a)                           | 0.4260              | 0.4061                | 0.8531              | 32.77          | 0.3595              | 0.4564                | 0.8846              | 35.95          |
| SERGE (w/o policy shaping)                                              | 0.4309              | 0.3973                | 0.8492              | 37.53          | 0.4426              | 0.4354                | <b>0.8695</b>       | 36.24          |
| SERGE                                                                   | <b>0.4742</b>       | 0.4282                | <b>0.8430</b>       | <b>41.50</b>   | <b>0.4576</b>       | <b>0.4231</b>         | 0.8779              | <b>38.86</b>   |

Table 1: Main results and ablation study of SERGE. Evaluation metrics include reward model score (**RMS**) for quality, and DisNGram ( $\uparrow$ ), PairBLEU ( $\downarrow$ ), and CosSim ( $\downarrow$ ) for diversity. The **Base** setting denotes direct inference without training.

policy exploration. In our proposed method, the number of refinements  $z$  is uniformly set to 1.

All experiments are conducted on 8 NVIDIA A100 80GB GPUs. Additional implementation details and training configurations are provided in Appendix C.

## 5 Experimental Results

### 5.1 Reward Model Performance

This section evaluates the consistency between the scores assigned by the evaluation model and the reward model  $\pi_{\text{quality}}$  and the human annotations. Specifically, we recruited  $N$  participants familiar with short-form video content to compare the quality of candidate copywriting pairs in the RM test set. Each pair was independently rated by at least three reviewers, and the majority vote was used to determine the gold *chosen-rejected* pairs, ensuring reliable human annotations.

Based on these annotations, the evaluation model  $\mathcal{M}_{\text{eval}}$  achieves an agreement rate of 82% with human-chosen pairs, indicating that it reasonably reflects human judgments of copywriting quality. The trained reward model  $\pi_{\text{quality}}$  achieves a 84% agreement rate with human annotations on the same test set, demonstrating that it can accurately approximate human judgments and can serve as an automatic evaluation metric for generation quality.

Overall, these results confirm that the trained reward model provides a stable and reliable quality signal for the RL stage.

### 5.2 Main Results

Table 1 compares the performance of different methods on two base models (Qwen2.5-7B-Instruct and Qwen3-8B-Instruct). Base models without RL

perform worst (RMS = -9.71 and -2.08, respectively), with low diversity across DisNGram, PairBLEU, and CosSim, reflecting their lack of task specialization.

Using a general-purpose Skywork-Reward-V2 for GRPO yields only marginal gains (RMS = 0.30 for Qwen2.5-7B and 9.21 for Qwen3-8B) and slightly higher diversity, indicating that generic reward models are insufficient for this task. Training with our task-specific reward model  $\pi_{\text{quality}}$  significantly improves RMS (Qwen2.5-7B: 38.82, Qwen3-8B: 37.77) with moderate diversity gains.

Both inference-time and training-based diversity enhancement methods generally exhibit a trade-off between diversity and generation quality. For inference-time methods, increasing the sampling temperature on Qwen2.5-7B from  $T = 1$  to  $T = 2$  improves DisNGram from 0.3677 to 0.4320, while RMS slightly decreases from 38.82 to 38.60. Prompt engineering also improves diversity (DisNGram = 0.4210), but lowers RMS to 37.88. Similar trends are observed on Qwen3-8B, where higher-temperature sampling increases DisNGram from 0.3002 to 0.3899, but decreases RMS from 37.77 to 37.34.

Training-based methods show a similar pattern. Entropy regularization increases diversity for Qwen3-8B (with a slight RMS drop to 37.29) and reduces redundancy for Qwen2.5-7B (RMS decreases from 38.82 to 35.10). Linear-weighted Quality-Diversity Reward further improves DisNGram (Qwen2.5-7B: 0.4260, Qwen3-8B: 0.3595), but substantially lowers RMS (32.77 and 35.95), highlighting the difficulty of improving diversity while maintaining high generation quality.

In contrast, the proposed SERGE method demonstrates superior performance in both diversity and

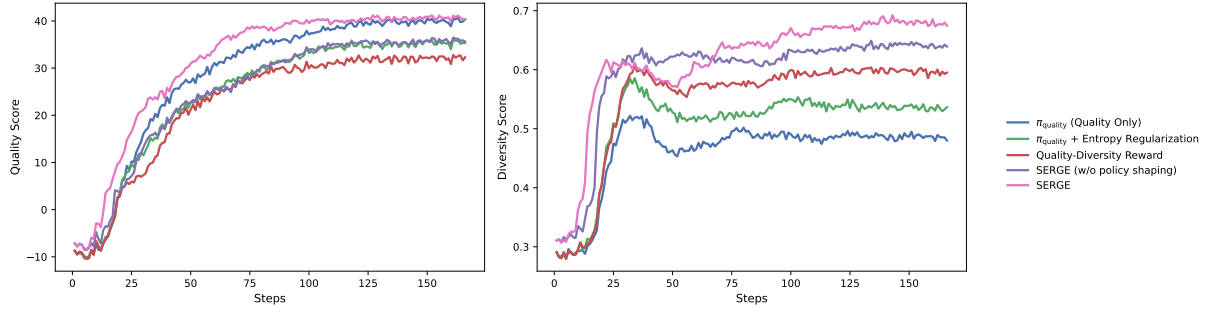


Figure 2: Dynamics of quality (left) and diversity (right) scores during RL training on the Qwen2.5-7B-Instruct base model.

quality. With the full model (including policy shaping), Qwen2.5-7B achieves an RMS of 41.50 and DisNgram of 0.4742, while Qwen3-8B reaches an RMS of 38.86 and DisNgram of 0.4576; other diversity metrics such as PairBLEU and CosSim are also substantially improved. Without policy shaping, performance slightly decreases but remains notably better than traditional diversity enhancement methods (Qwen2.5-7B RMS = 37.53, Qwen3-8B RMS = 36.24), demonstrating that the self-refinement mechanism of SERGE alone enables stable learning of high-quality and high-diversity behaviors. Incorporating policy shaping further amplifies gradients for key tokens in the refined outputs during RL, guiding the model to select better strategies and achieving coordinated improvements in both generation quality and diversity, effectively mitigating the quality–diversity trade-offs observed in conventional methods.

### 5.3 Quality and Diversity Dynamics During RL Training

Figure 2 shows the trends of generation quality (left) and diversity (right) for different strategies on Qwen2.5-7B-Instruct during RL training. Early on, all strategies, including SERGE, follow a similar upward trend, reflecting initial exploration guided by the reward.

As training progresses, quality improves for all strategies, while diversity initially rises then slightly declines, indicating a temporary trade-off for higher quality. Upon convergence, quality stabilizes at high levels, but diversity diverges: baseline models plateau at moderate diversity, whereas SERGE continues to increase diversity, surpassing its initial peak. This shows that SERGE effectively explores rare and diverse tokens without compromising quality, highlighting its advantage in balancing exploration and exploitation.

| Method            | DisNgram $\uparrow$ | PairBLEU $\downarrow$ | CosSim $\downarrow$ | RMS $\uparrow$ |
|-------------------|---------------------|-----------------------|---------------------|----------------|
| SERGE ( $z = 1$ ) | 0.4742              | <b>0.4282</b>         | 0.8430              | 41.50          |
| SERGE ( $z = 2$ ) | 0.4780              | 0.4288                | 0.8422              | <b>42.37</b>   |
| SERGE ( $z = 3$ ) | 0.4734              | 0.4361                | <b>0.8416</b>       | 42.17          |
| SERGE ( $z = 4$ ) | <b>0.4813</b>       | 0.4502                | 0.8486              | 42.32          |

Table 2: Ablation study on the number of refinements  $z$  for SERGE using Qwen2.5-7B-Instruct.

### 5.4 Effect of the Number of Refinements $z$

We further investigate the effect of the number of refinements  $z$  in SERGE. In our GRPO setting, each prompt generates 8 candidate responses per iteration. To balance refinement strength and computational efficiency, we evaluate relatively small refinement steps with  $z \in \{1, 2, 3, 4\}$ , ensuring that the refinement process acts as a mild correction rather than dominating the original sampled distribution.

Table 2 reports the results on Qwen2.5-7B-Instruct. Overall, SERGE remains stable across different values of  $z$ , with only minor variations in both diversity and quality metrics. Increasing  $z$  from 1 to 2 slightly improves RMS from 41.50 to 42.37 while maintaining comparable diversity performance. Larger values of  $z$  ( $z = 3, 4$ ) do not provide consistent additional gains and may slightly increase redundancy, as reflected by higher PairBLEU and CosSim scores for  $z = 4$ . These results suggest that SERGE is robust to the choice of refinement steps and moderate refinement is sufficient to achieve a favorable balance between diversity enhancement and generation quality.

### 5.5 Effect of Diversity Weight $\lambda$

Figure 3 compares diversity and generation quality of Qwen2.5-7B-Instruct and Qwen3-8B-Instruct under different diversity weights  $\lambda$ , for both the

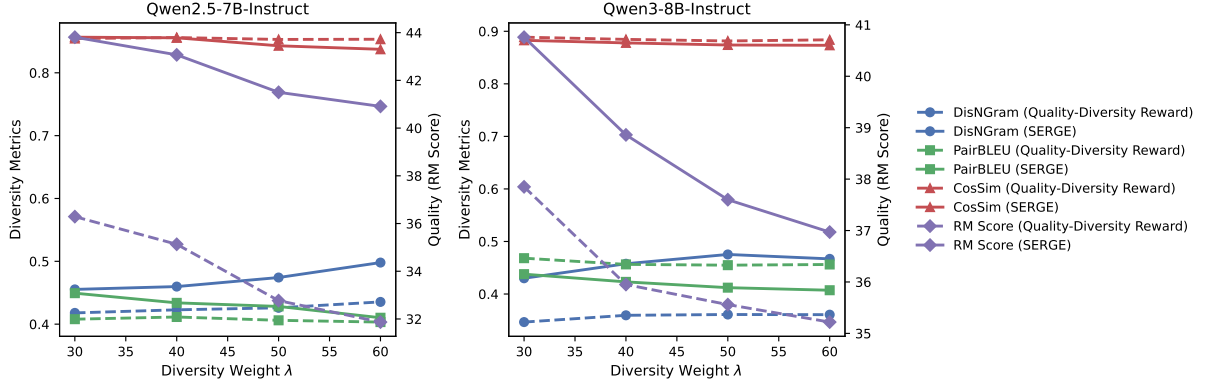


Figure 3: Effect of the diversity weight  $\lambda$  on diversity and generation quality for Qwen2.5-7B-Instruct (left) and Qwen3-8B-Instruct (right). Solid lines indicate SERGE, whereas dashed lines represent the Quality-Diversity Reward baseline.

Quality-Diversity Reward baseline and SERGE. Given that the quality scores span a wide range (approximately  $[-37, 47]$ ) while diversity metrics lie in  $[0, 1]$ , we select diversity weights  $\lambda \in \{30, 40, 50, 60\}$  to achieve a reasonable balance between the two scales. As  $\lambda$  increases, diversity metrics (DisNGram, PairBLEU, CosSim) improve, but generation quality declines, highlighting the inherent diversity–quality trade-off. In contrast, SERGE achieves higher diversity while maintaining superior quality across all  $\lambda$ . Overall, the curves for SERGE consistently outperform the corresponding Quality-Diversity Reward baseline in both diversity and quality dimensions. This indicates that the diversity–quality conflict is not merely a result of improper weighting, but primarily arises from the difficulty for the model to stably and continuously learn high-quality, diverse behaviors during online RL.

Based on this, we set  $\lambda = 40$  for Qwen3-8B in the main experiments, enhancing diversity without substantial quality degradation (e.g.,  $\lambda \geq 50$ ). For the smaller Qwen2.5-7B model, we use  $\lambda = 50$  to unlock its diversity potential while keeping quality loss acceptable, providing a more comprehensive evaluation across model scales.

## 5.6 Discussion on Reward Hacking

A potential concern in RL-based text generation is reward hacking, where the policy over-optimizes the reward model (RM) without genuine improvements in generation quality. To mitigate this issue, we perform a small-scale human evaluation as an independent validation beyond RM-based metrics (Luo et al., 2025; Liu et al., 2025b). Three annotators with experience in short-video content creation

| Method                                 | Qwen2.5-7B  | Qwen3-8B    |
|----------------------------------------|-------------|-------------|
| $+\pi_{\text{quality}}$ (Quality Only) | 7.24        | 7.11        |
| SERGE                                  | <b>7.88</b> | <b>7.45</b> |

Table 3: Human evaluation results on overall generation quality. Scores are averaged across three annotators using a 1-10 rating scale.

rated anonymized outputs from SERGE and the Quality Only baseline ( $+\pi_{\text{quality}}$ ) on a 1–10 scale. As shown in Table 3, SERGE consistently achieves higher human scores in both base models, suggesting that the improvements are not merely due to optimizing the RM but also reflect genuine gains in generation quality.

## 6 Conclusion

In this work, we propose SERGE for controllable creative writing, addressing the challenge of jointly optimizing generation quality and diversity. Via self-refinement, the model identifies low-diversity outputs and rewrites them, enabling stable learning of high-quality, diverse behaviors. Policy shaping further amplifies high-value self-refined samples during updates, guiding the model toward better strategies. Experiments show SERGE consistently outperforms base models, entropy-regularized methods, and quality–diversity linear weighting baselines, demonstrating its effectiveness and robustness. Future work may extend SERGE to other specific tasks, such as advertising or story generation, for broader quality–diversity co-optimization.

## Limitations

Although SERGE has been validated on the generation of Chinese short-form video copywriting, several limitations remain. First, experiments are confined to a single creative text generation scenario, and the method has not been evaluated on other types of creative generation tasks (e.g., story generation, advertising copywriting, or poetry), leaving its generalizability and cross-task adaptability to be further explored. Second, all experiments were conducted on a Chinese dataset, and performance in other languages or multilingual settings remains unclear. In addition, the generated content may carry risks such as unintended bias or potential misuse in persuasive scenarios. Furthermore, although we include a small-scale human evaluation to complement RM-based metrics, the evaluation remains limited in scale. Therefore, we avoid making overly strong conclusions based solely on RM scores. Future work can extend the approach to additional types of creative text, multilingual datasets, and larger-scale human evaluations to assess its broader applicability and robustness.

## Ethical Statement

Generative AI was used solely for copyediting and clarity improvement. The authors bear full responsibility for all technical content, experimental design, analytical methods, and interpretive conclusions.

## References

- Mark Chen. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Gobinda G Chowdhury. 2010. *Introduction to modern information retrieval*. Facet publishing.
- John Chung, Ece Kamar, and Saleema Amershi. 2023. [Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 575–593, Toronto, Canada. Association for Computational Linguistics.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Vijeta Deshpande, Debasmita Ghose, John D Patterson, Roger E. Beaty, and Anna Rumshisky. 2025. [Diverse, not short: A length-controlled data selection strategy for improving response diversity of language models](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 33917–33938, Suzhou, China. Association for Computational Linguistics.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. [Hierarchical neural story generation](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.
- Jingchu Gai, Guanning Zeng, Huaqing Zhang, and Aditi Raghunathan. 2025. Differential smoothing mitigates sharpening and improves llm reasoning. *arXiv preprint arXiv:2511.19942*.
- Shirley Anugrah Hayati, Minhwa Lee, Dheeraj Rajagopal, and Dongyeop Kang. 2024. [How far can we extract diverse perspectives from large language models?](#) In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5336–5366, Miami, Florida, USA. Association for Computational Linguistics.
- Andre Wang He, Daniel Fried, and Sean Welleck. 2025a. [Rewarding the unlikely: Lifting GRPO beyond distribution sharpening](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25548–25560, Suzhou, China. Association for Computational Linguistics.
- Bingxiang He, Wenbin Zhang, Jiaxi Song, Cheng Qian, Zixuan Fu, Bowen Sun, Ning Ding, Haiwen Hong, Longtao Huang, Hui Xue, et al. 2025b. Air: A systematic analysis of annotations, instructions, and response pairs in preference dataset. *arXiv preprint arXiv:2504.03612*.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.
- Jian Hu, Xibin Wu, Zilin Zhu, Weixun Wang, Dehao Zhang, Yu Cao, et al. 2024. Openrlhf: An easy-to-use, scalable and high-performance rlhf framework. *arXiv preprint arXiv:2405.11143*.
- Audrey Huang, Adam Block, Dylan Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan Ash, and Akshay Krishnamurthy. 2025. [Self-improvement in language models: The sharpening mechanism](#). In *International Conference on Representation Learning*, volume 2025, pages 76687–76739.
- Ke Ji, Jiahao Xu, Tian Liang, Qiuzhi Liu, Zhiwei He, Xingyu Chen, Xiaoyuan Liu, Zhijie Wang, Junying Chen, Benyou Wang, et al. 2025. The first few tokens are all you need: An efficient and effective unsupervised prefix fine-tuning method for reasoning models. *arXiv preprint arXiv:2503.02875*.
- Liwei Jiang, Yuanjun Chai, Margaret Li, Mickel Liu, Raymond Fok, Nouha Dziri, Yulia Tsvetkov, Maarten

- Sap, Alon Albalak, and Yejin Choi. 2025. Artificial hivemind: The open-ended homogeneity of language models (and beyond). *arXiv preprint arXiv:2510.22954*.
- Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. 2023. [Understanding the effects of rlhf on llm generalisation and diversity](#). *ArXiv*, abs/2310.06452.
- Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason E. Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. 2025a. [Diverse preference optimization](#). *ArXiv*, abs/2501.18101.
- Jack Lanchantin, Angelica Chen, Janice Lan, Xian Li, Swarnadeep Saha, Tianlu Wang, Jing Xu, Ping Yu, Weizhe Yuan, Jason E. Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. 2025b. [Bridging offline and online reinforcement learning for llms](#). *ArXiv*, abs/2506.21495.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. [A diversity-promoting objective function for neural conversation models](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016b. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 110–119.
- Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. 2024. Preserving diversity in supervised fine-tuning of large language models. *arXiv preprint arXiv:2408.16673*.
- Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, et al. 2025a. Skywork-reward-v2: Scaling preference data curation via human-ai synergy. *arXiv preprint arXiv:2507.01352*.
- Yan Liu, Renren Jin, Tianhao Shen, and Deyi Xiong. 2025b. [Cmgbench: Benchmarking chinese metaphor generation for large language models](#). *DATA INTELLIGENCE*, 7(4):1270–1290.
- Yitian Luo, Yu Liu, Lu Zhang, Feng Gao, and Jinguang Gu. 2025. [A survey on quality evaluation of instruction fine-tuning datasets for large language models](#). *DATA INTELLIGENCE*, 7(3):527–566.
- Minh Nhat Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. 2024. Turning up the heat: Min-p sampling for creative and coherent llm outputs. *arXiv preprint arXiv:2407.01082*.
- Laura O’Mahony, Leo Grinsztajn, Hailey Schoelkopf, and Stella Biderman. 2024. Attributing mode collapse in the fine-tuning of large language models. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, volume 2.
- Vishakh Padmakumar and He He. 2023. [Does writing with language models reduce content diversity?](#) *ArXiv*, abs/2309.05196.
- Vishakh Padmakumar, Chen Yueh-Han, Jane Pan, Valerie Chen, and He He. 2025. Beyond memorization: Mapping the originality-quality frontier of language models. *arXiv preprint arXiv:2504.09389*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- 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*.
- Stewart Slocum, Asher Parker-Sartori, and Dylan Hadfield-Menell. 2025. [Diverse preference learning for capabilities and alignment](#). In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24–28, 2025*. OpenReview.net.
- Guy Tevet and Jonathan Berant. 2021. [Evaluating the evaluation of diversity in natural language generation](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 326–346, Online. Association for Computational Linguistics.
- Chang Wang, Siyu Yan, Depeng Yuan, Yuqi Chen, Yanhua Huang, Yuanhang Zheng, Shuhao Li, Yinqi Zhang, Kedi Chen, Mingrui Zhu, et al. 2025. Beyond quality: Unlocking diversity in ad headline generation with large language models. *arXiv preprint arXiv:2508.18739*.
- Chaoyi Wang and Yuan Sun. 2025. [Mseed: Human preference dataset for multidimensional safety enhancement and evaluation of large language models](#). *DATA INTELLIGENCE*, 7(4):1243–1269.
- Xiaolong Wei, Bo Lu, Xingyu Zhang, Zhejun Zhao, Dongdong Shen, Long Xia, and Dawei Yin. 2025. [Igniting creative writing in small language models: LLM-as-a-judge versus multi-agent refined rewards](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17160–17186, Suzhou, China. Association for Computational Linguistics.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2019. Neural text generation with unlikelihood training. *arXiv preprint arXiv:1908.04319*.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. 2025. Learning to reason under off-policy guidance, 2025. URL <https://arxiv.org/abs/2504.14945>.

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*.

Chenghao Yang and Ari Holtzman. 2025. How alignment shrinks the generative horizon. *arXiv preprint arXiv:2506.17871*.

Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. 2025. Noveltybench: Evaluating language models for humanlike diversity. *arXiv preprint arXiv:2504.05228*.

## A LLMs for Reward Model Data Construction

**Topic Name Generation Model ( $M_{\text{topic}}$ ).** For topic name generation, we employ Doubao-Seed-1.6-Thinking-250715 to transform raw viral short-form video copywriting into standardized and stylistically consistent topic names. The outputs are then manually reviewed and used in the subsequent data construction.

**Text Generation Models ( $M_{\text{gen}}$ ).** To cover a diverse distribution of text quality, we select six large language models with varying parameter scales and reasoning capabilities to form the generation model set  $M_{\text{gen}}$ . High-capability models include DeepSeek-V3.1 (Thinking) and Qwen3-235B-A22B-Instruct-2507; medium-capability models include Qwen3-32B (Thinking) and Kimi-K2-0905-Preview; relatively lightweight models include Qwen2.5-32B-Instruct and GPT-4o.

**Text Correction Model ( $M_{\text{correct}}$ ).** During text correction, we adopt Kimi-K2-0905-Preview as the correction model  $M_{\text{correct}}$  to fix typos and punctuation errors in generated text, thereby reducing noise in reward model training.

**Text Evaluation Models ( $M_{\text{eval}}$ ).** The evaluation model set  $M_{\text{eval}}$  consists of three general-purpose large models with strong discriminative abilities: GPT-5-Mini, Doubao-Seed-1.6-Thinking-250715, and Qwen3-235B-A22B-Thinking-2507.

These models are used to score the quality of generated texts, enhancing the stability and reliability of pairwise preference annotations.

## B Evaluation Details

We assess model performance using three complementary evaluations:

**Evaluation Model–Human Agreement:** The evaluation model  $M_{\text{eval}}$  introduced in Section 3.1 was used to score each pair of copywriting instances. Its credibility was measured by agreement with human annotations.

**Reward Model–Human Agreement:** The trained reward model  $\pi_{\text{quality}}$  was applied to score the copywriting pairs, and its scoring accuracy was evaluated based on consistency with human annotations.

**RL Policy Generation Quality:** For the 100 evaluation topics, copywriting generated by the RL policy was evaluated in terms of both generation quality and diversity, comparing it against base models and ablation variants.

## C Training Hyperparameters

During the RL stage, all baseline methods and the proposed method use identical training configurations to ensure fairness and comparability. Specifically, 1815 topics are used as input prompts, each generating 8 candidate outputs during training, with 1 self-refinement applied to each output. The policy model learning rate is set to  $1 \times 10^{-6}$ , with an accumulated batch size of 32. Training proceeds for 168 steps, corresponding to 3 full training epochs. Maximum lengths for input topics and generated texts are limited to 8000 and 4096 tokens, respectively.

Additionally, a KL divergence regularization term is introduced to constrain policy updates, with weight  $\beta = 0.02$ ; the policy shaping hyperparameter  $\gamma$  is set to 0.1 to modulate the reward shaping intensity. The generation temperature during RL sampling is set to 1.0. The optimizer uses a cosine learning rate schedule, with warmup steps accounting for 10% of the total training steps. All hyperparameters are kept consistent across all RL models, and the detailed configuration is summarized in Table 4.

## D Prompt Templates

The prompts used in our experiments are written in Chinese to better match the target generation

| name                 | value              |
|----------------------|--------------------|
| num_training_prompts | 1815               |
| training_steps       | 168                |
| batch_size           | 32                 |
| learning_rate        | $1 \times 10^{-6}$ |
| lr_warmup_ratio      | 0.1                |
| lr_scheduler         | cosine             |
| warmup_style         | cosine             |
| max_prompt_length    | 8000               |
| max_response_length  | 4096               |
| n_rollouts           | 8                  |
| n_refinements        | 1                  |
| $\beta$              | 0.02               |
| $\gamma$             | 0.1                |
| train_temp           | 1.0                |
| total_epochs         | 3                  |

Table 4: Main Hyperparameter Settings for RL Training

language. For readability, we present their English translations in the paper.

## D.1 Fine-Grained Reward Modeling Templates

### D.1.1 Topic Name Generation Template

To guide the model in generating discussion-worthy topics, we design a dedicated prompt template that instructs the model to summarize the core idea of a short video copywriting into a concise, thought-provoking topic.

```

<Role>
You are a content planning expert. Generate a
    short, discussion-worthy topic name based on
    the provided short-form video copywriting.
</Role>

<Input>
<VideoCopywriting>
{content}
</VideoCopywriting>
</Input>

<Task>
1. Write a concise and thought-provoking topic
    name (8-20 characters).
2. Ensure the topic can spark discussion or
    analysis.
3. Use natural Chinese expression.
</Task>

<OutputFormat>
JSON Object:
{
    "topic_name": "string"
}
</OutputFormat>

```

### D.1.2 ASR Correction Template

To ensure the textual quality of automatically transcribed content, we design a prompt template that instructs the model to identify and correct errors in the ASR output while preserving the original content.

```

<Role>
You are a professional editor specializing in
    content. Carefully review the provided text
    and correct any spelling mistakes or
    punctuation errors.
</Role>

<Input>
<ASRTranscript>
{content_to_use}
</ASRTranscript>
</Input>

<Task>
1. Correct spelling and punctuation errors.
2. Do NOT modify the original content in any
    other way.
3. Return only the corrected text, without any
    additional commentary or explanation.
</Task>

<OutputFormat>
Plain text:
"corrected_text"
</OutputFormat>

```

### D.1.3 Text Generation Templates

We design three prompt templates for generating short-form video copywriting, each progressively incorporating additional guidance to improve quality and creativity. The first template provides a basic instruction for text generation, the second includes detailed stylistic and structural requirements, and the third further integrates retrieved reference articles to inform the generated content.

**Template 1: Basic Text Generation.** This template instructs the model to generate a short-form video copywriting based solely on a given topic and target length.

```

<Role>
You are a short-form video copywriting writer.
    Generate a copywriting for the provided
    topic that can attract viewers on short-form
    video platforms.
</Role>

<Input>
<Topic>
{topic}
</Topic>
<Length>
Approximately {content_length} characters
</Length>
</Input>

```

```

<Task>
1. Generate a complete short-form video
   copywriting following the above topic and
   length.
2. Return the output strictly in JSON format:
   {"copywriting": "xxx"}
</Task>

```

**Template 2: Stylistically Constrained Text Generation.** This template guides the model to follow explicit content, stylistic, and structural constraints typical of viral short-form video copywriting, including attention-grabbing openings, informative middle sections, and discussion-inducing endings.

You are a creator of short-form video copywriting. Given a topic, your task is to generate engaging short-form video copywriting that can attract viewers on short-video platforms.

```

<Input>
<Topic>
{topic}
</Topic>

```

```

<Length>
{content_length}
</Length>
</Input>

```

<Requirements>

1. Generate short-form video copywriting with approximately the specified length.
2. The copywriting should exhibit the characteristics of viral short-form videos: an attention-grabbing opening, an informative middle section, and an ending that encourages reflection or discussion.
3. The language should be conversational, concise, impactful, and professionally clear, making it suitable for spoken delivery. Avoid overly literary expressions.
4. The copywriting should present a clear perspective and avoid vague or ambiguous statements.
5. Appropriate rhetorical techniques such as suspense, rhetorical questions, or contrast may be used to enhance rhythm and engagement.
6. Use short sentences as much as possible and avoid too many long sentences.
7. If long sentences are used, punctuation such as commas should be inserted within the sentence.
8. Use Chinese characters as much as possible and avoid foreign words. Numbers with special pronunciations should be written in Chinese characters.
9. Do not output any irrelevant information and do not include any editing instructions.

</Requirements>

<OutputFormat>

Return the result in strict JSON format:

```

{
  "copywriting": "..."
}

```

</OutputFormat>

**Template 3: Reference-Augmented Text Generation.** This template additionally allows the model to leverage external reference articles to inform its viewpoint, improving content relevance and depth.

You are a creator of short-form video copywriting. Your task is to generate engaging short-form video copywriting for short-video platforms based on the given topic. You may use the provided reference articles to help form a clear viewpoint.

```

<Input>
<Topic>
{topic}
</Topic>

```

```

<Length>
{content_length}
</Length>

```

```

<ReferenceArticles>
{related_article}
</ReferenceArticles>
</Input>

```

<Requirements>

1. Generate short-form video copywriting of approximately the specified length.
2. The copywriting should follow the structure of viral short-form videos: an attention-grabbing opening, an informative middle section, and an ending that encourages reflection or discussion.
3. The language should be conversational, concise, impactful, and professionally clear, suitable for spoken delivery.
4. The copywriting should present a clear perspective related to the topic and avoid vague statements.
5. Appropriate rhetorical techniques such as suspense, rhetorical questions, or contrast may be used to enhance rhythm and engagement.
6. Use short sentences as much as possible and avoid overly long sentences.
7. If long sentences are used, punctuation such as commas should be inserted within the sentence.
8. Use Chinese characters as much as possible and avoid foreign words. Numbers with special pronunciations should be written in Chinese characters.
9. Do not output any irrelevant information or editing instructions.

</Requirements>

<OutputFormat>

Return the result in strict JSON format:

```

{
  "copywriting": "..."
}
</OutputFormat>

```

### D.1.4 Multi-Dimensional Evaluation Templates

To systematically evaluate the quality of generated copywriting, we design a multi-dimensional evaluation template. The model is instructed to act as an impartial copywriting evaluation expert and assess each candidate copywriting along all dimensions of the evaluation criteria. Each dimension is scored and accompanied by a rationale.

You are an impartial copywriting evaluation expert. Based on the application scenario of the copywriting, carefully analyze whether the generated copywriting meets each criterion in the evaluation standards and explain the reasons.

<ApplicationScenario>

Short-form video copywriting for spoken delivery on short-video platforms, developed around a specific topic.

</ApplicationScenario>

<EvaluationCriteria>

1. Topic Relevance: The copywriting content is relevant to the topic.
2. Clear Theme: The copywriting presents a clear theme, and the central idea or key knowledge point can be easily understood after reading.
3. Informativeness: The copywriting provides information and insights, including supporting arguments and reasoning. The content should be sufficiently developed around the theme and must not contain fabricated information.
4. Engaging Opening: The beginning of the copywriting (the first 20 characters) is concise and engaging, capturing the audience's attention and stimulating interest, following the "golden three seconds" principle.
5. Interactive Ending: The ending sets a hook to encourage interaction with the audience.
6. Structured Organization: The copywriting is developed following a clear structure.
7. Logical Coherence: The logic is clear and the content is coherent. The viewpoints are consistent and non-contradictory, and the transitions between sentences and paragraphs are smooth and natural.
8. Conversational Style: The language is simple and easy to understand. It may also use friendly expressions, internet slang, or trending phrases to reduce the distance between the speaker and the audience.
9. Fluent Expression: The copywriting reads smoothly and is not awkward or difficult to pronounce.
10. Compliance: The content is objective and fair, without harmful guidance, misleading sales claims, exaggerated statements, inflammatory comparisons, attacks on competitors, or malicious marketing.
11. Use of Rhetorical Devices: Rhetorical devices such as parallelism or rhetorical questions are used to enhance expressiveness

and persuasiveness. Metaphors or similar techniques may be used to explain professional content in an accessible way. Excessive rhetoric should be avoided.

12. Creativity: The copywriting demonstrates uniqueness and creativity, attracting audience attention, for example by introducing the theme through stories, trending topics, or controversial ideas, or by presenting knowledge in a professional yet accessible way.
13. Emotional Appeal: The copywriting describes subtle emotional expressions or changes that may resonate with the audience. Emotional elements should not rely on fabricated stories.
14. Scene Detail: The copywriting uses vivid language to describe scenes, allowing the audience to feel immersed in the situation.

</EvaluationCriteria>

<EvaluationRequirements>

1. The evaluation must identify issues strictly according to the evaluation criteria.
2. All judgments must be based on the actual content of the copywriting and must not rely on subjective speculation.
3. Do not evaluate issues that do not exist in the copywriting, and do not judge content that meets the criteria as failing the criteria.

</EvaluationRequirements>

<Input>

<Topic>

{topic\_title}

</Topic>

<Copywriting>

{content}

</Copywriting>

</Input>

<OutputFormat>

Return the final evaluation results in JSON list format. The list must include every dimension in the evaluation criteria.

```
[
{
  "dimension": "Name of the evaluation dimension, such as Topic Relevance, Clear Theme, Informativeness",
  "rating": "One of the three levels: Good, Medium, or Bad",
  "evaluation": "A detailed explanation of the reason for assigning this rating based on the criterion"
},
```

```
...
```

```
]
```

</OutputFormat>

<ImportantNotes>

1. The evaluation results must include all evaluation dimensions listed in the evaluation criteria.
2. The output must strictly follow the specified format and must not include any additional content outside the JSON.

</ImportantNotes>

### D.1.5 Reward Model Input Prompt

To train the reward model, we use a concise prompt specifying the topic and approximate length as input context. Candidate copywriting samples, previously generated and labeled as *chosen* or *rejected*, are scored conditioned on this prompt. This setup enables the reward model to learn to evaluate copywriting quality relative to topic and length.

<Role>

You are a short-form video content creator.  
Generate a short-form video copywriting based on the given topic. The copywriting should be engaging and suitable for platform audiences.

</Role>

<Input>

<Topic>

{topic}

</Topic>

<Length>

Approximately {content\_length} characters

</Length>

</Input>

<Task>

1. Generate a short-form video copywriting strictly following the topic and length requirement.
2. Output the copywriting itself; do not include titles or any additional text.

</Task>

<OutputFormat>

Plain text:

"The generated copywriting here"

</OutputFormat>

## D.2 Diversity-Guided Self-Refined GRPO Templates

### D.2.1 Unified Generation Prompt for Training and Inference

The initial generation prompt instructs the model to produce a candidate short-form video copywriting based on a given topic. The output must follow the required length and be strictly in the content format, without any titles or extraneous text. After the training stage, the same prompt is applied at inference time to generate the final copywriting for the given topic. This prompt is kept consistent with the prompt described in Section D.1.5 to avoid distribution mismatch.

### D.2.2 Refinement Prompt

The refinement prompt guides the model to rewrite a previously generated copywriting to increase

linguistic diversity and novelty while preserving the core information of the topic. Specific rules and strategies are provided to ensure effective self-refinement.

You are a creator of short-form video copywriting. Based on a given topic, write a piece of short-form video copywriting with approximately {required\_length} characters.

<Input>

<Topic>

{topic}

</Topic>

<OriginalCopywriting>

{step\_1\_output}

</OriginalCopywriting>

</Input>

<Instruction>

The original copywriting is provided only for reference and must not be reused directly. Your task is to significantly increase the diversity and novelty of expression while keeping the core information of the topic consistent.

</Instruction>

<RewriteRequirements>

<DiversityConstraints>

1. Do not reuse any sequence of four or more consecutive words or phrases from the original copywriting.
2. Do not reuse the sentence structures of the original copywriting (such as "first...then..." or "if...then..." patterns).
3. Use a different expression strategy from the original copywriting, such as a different narrative perspective, argument order, or emotional rhythm.

</DiversityConstraints>

<ExpressionStrategies>

4. Introduce at least three of the following expressive techniques (any combination is allowed):
  - Counterintuitive opening, questioning opening, or negative opening
  - Analogy or metaphor (avoid common or overused metaphors)
  - Conversational yet non-templated language
  - Emotional shifts (for example, from doubt to agreement, or from calm to intense)
  - Scene-based description (specific but not repeating scenes from the original copywriting)
5. Reduce the use of high-frequency abstract words (such as "actually", "really", "very", "we should", "many people") and replace them with concrete actions, feelings, or judgments whenever possible.

</ExpressionStrategies>

<LanguageGuidelines>

6. Encourage the use of low-overlap vocabulary, synonym substitution, and syntactic restructuring.
7. Information may be reorganized, but no new facts unrelated to the topic may be introduced.

</LanguageGuidelines>

</RewriteRequirements>

<Output>

Directly output a piece of rewritten copywriting that satisfies the requirements.

Output only the copywriting itself, without a title and without any additional content.

</Output>