From Coarse to Fine: Benchmarking and Reward Modeling for Writing-Centric Generation Tasks

Qingyu Ren[1], Tianjun Pan[1], Xingzhou Chen[1], Xuhong Wang[2]*
[1]Shanghai Key Laboratory of Data Science,
College of Computer Science and Artificial Intelligence, Fudan University,
[2]Shanghai Artificial Intelligence Laboratory
{qyren24,tjpan24, xzchen24}@m.fudan.edu.cn, wangxuhong@pjlab.org.cn

Abstract

Large language models have achieved remarkable progress in text generation but still struggle with generative writing tasks. In terms of evaluation, existing benchmarks evaluate writing reward models coarsely and fail to measure performance from the perspective of specific requirements. In terms of training, existing training methods either use LLM-as-a-judge approaches or train coarse-grained reward models, lacking fine-grained requirement-adherence reward modeling. To address these issues, we propose a fine-grained evaluation pipeline WEval for writing reward models and a fine-grained reinforcement learning training framework WRL. The evaluation data of WEval covers multiple task categories and requirement types, enabling systematic evaluation of writing reward models by measuring the correlation between the rankings of the reward model and gold rankings. WRL constructs positive and negative samples by selectively dropping instruction requirements, allowing for more precise reward model training. Experiments show that our models achieve substantial improvements across various writing benchmarks and exhibit strong generalization. The code and data are publicly available at https://github.com/Rainier-rq1/From_Coarse_to_Fine.

1 Introduction

Large language models have made significant progress in text generation (Nagano et al., 2025; Sen et al., 2025; Wu et al., 2025a). However, generative writing tasks present significant challenges for these models, such as creative writing (Liao et al., 2025; Wei et al., 2025; Li et al., 2026), story generation (Wang and Kreminski, 2024; Venkatraman et al., 2025; Liu et al., 2026), and report generation (Ding et al., 2024; Wang et al., 2025; Yuan et al., 2024a). These tasks require models to have high-level comprehension, logical reasoning (Xu et al., 2026; Cheng, 2026), and instruction-following capabilities (Bai et al., 2024; Wu et al., 2025c; Ren et al., 2025a; He et al., 2024).

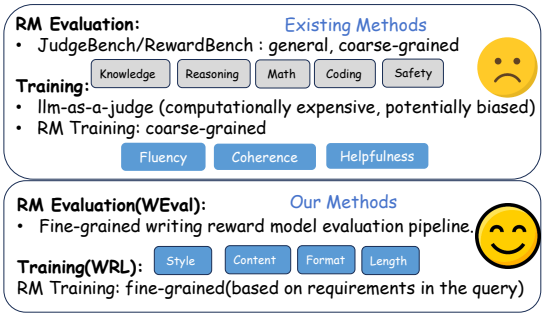

Figure 1: Comparison with prior writing-task evaluation and training paradigms.

The instructions in writing tasks often contain many specific requirements, and the model's output needs to satisfy those requirements. For improving the writing ability of models, current training paradigms can be categorized into supervised fine-tuning and reinforcement learning. Both supervised fine-tuning (SFT) and direct preference optimization (DPO) rely on high-quality outputs or preference pairs, which are difficult to obtain (Pham et al., 2024; Bai et al., 2024; Ren et al., 2025b). The reinforcement learning with verifiable rewards (RLVR) training paradigm addresses this problem, requiring only instructions as input and verifiable rewards for reinforcement learning (Wu et al., 2025b,c; Chen et al., 2025).

Current evaluation and RLVR training paradigms for writing tasks have the following limitations, as shown in Fig. 1. **In terms of evaluation**, current benchmarks provide coarse-grained, general evaluation of writing reward models across dimensions like knowledge, reasoning, and safety (Tan et al., 2024; Lambert et al., 2025; Huang et al., 2026), but fail to offer fine-grained assessment of reward modeling performance on

* Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 2796–2810
July 2-7, 2026

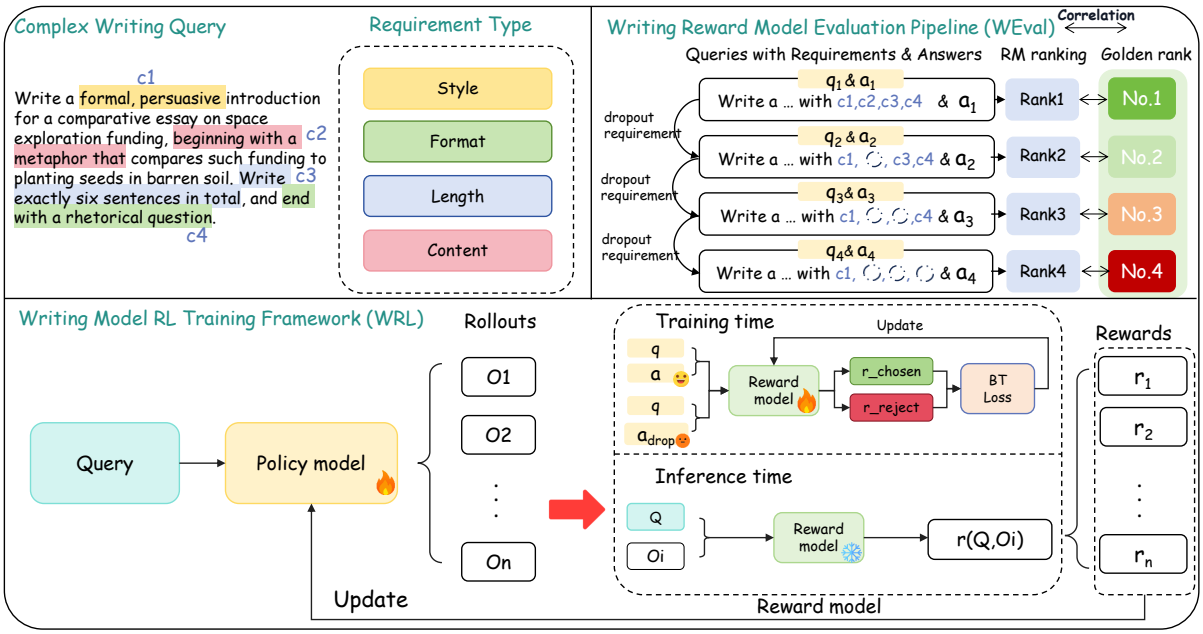

Figure 2: Our framework comprises two components: (WEval), an evaluation pipeline that applies requirement dropout to construct a partial order over requirement adherence, from which the golden rankings are derived, and evaluates reward models by measuring their correlation with the golden rankings; and (WRL), a reinforcement learning framework with fine-grained Bradley-Terry training for reward models, which provide reward signals for RL training.

specific requirements in writing tasks. **In terms of training paradigms**, existing methods either provide rewards using a strong reasoning model in the LLM-as-a-judge manner (Chen et al., 2025; Peng et al., 2025), which is computationally expensive, potentially biased, and overdependent on the evaluation LLM's capability, or train a coarse-grained reward model based on attributes such as fluency, coherence, and helpfulness (Wu et al., 2025b; Yuan et al., 2024b). These approaches lack fine-grained scoring based on requirement adherence, which limits the effectiveness of RLVR training.

To address these issues, we propose a fine-grained evaluation pipeline WEval for writing reward models and a reinforcement learning training framework WRL shown in Fig. 2. Our approach applies dropout to requirements in complex writing queries to construct positive and negative examples, where more requirement dropouts result in worse requirement adherence, thereby naturally forming the golden rankings. We evaluate reward models by comparing their rankings with these golden rankings through correlation metrics. Furthermore, the positive and negative examples obtained via requirement dropout enable fine-grained Bradley-Terry (BT) training for writing reward models, which precisely scores model responses based on requirement adherence for effective RLVR training.

In summary, our contributions are as follows: (1) We propose a fine-grained evaluation pipeline WEval for writing reward models; (2) We introduce a fine-grained training framework WRL; (3) We demonstrate that the trained writing reward models effectively enhance the writing capabilities of policy models through reinforcement learning training, and that the writing reward model evaluation results are consistent with the RL training performance across different reward models.

2 Related Work

2.1 Reward Model Evaluation Benchmarks

Existing reward model evaluation benchmarks suffer from coarse-grained evaluation dimensions that fail to assess performance on specific requirements. RewardBench (Lambert et al., 2025) evaluates reward models across general dimensions (Chat, Chat Hard, Safety, Reasoning) but only measures overall accuracy on pairwise comparisons, without decomposing complex instructions into specific requirement types. Similarly, JudgeBench (Tan et al., 2024) evaluates LLM-as-a-Judge systems across dimensions (decision-making ability, consistency, fairness, robustness) but these dimensions are too coarse-grained to systematically evaluate

reward models' ability to distinguish between responses with different levels of requirement adherence. Both benchmarks rely on binary pairwise judgments and task-level evaluation, which cannot capture the nuanced ranking quality needed for fine-grained writing evaluation. Different from these benchmarks, our evaluation pipeline enables fine-grained assessment by comparing reward models' rankings of multiple candidate responses with naturally formed golden rankings, systematically evaluating the quality of writing reward models on specific requirements.

2.2 Training Methods for Writing Improvement

Existing training methods for writing models can be divided into supervised fine-tuning and reinforcement learning (Wu et al., 2025c; Pham et al., 2024). SFT training focuses on building high-quality datasets of instruction–response pairs for writing tasks (Quan et al., 2024). LongWriter (Bai et al., 2024) is trained in stages using SFT and DPO (Rafailov et al., 2023) to enhance long-text quality. Recently, many researchers have trained writing models using RLVR. ACE-RL (Chen et al., 2025) employs Qwen3-8B-as-a-judge for constraint verification—which relies heavily on external large models and leads to high computational costs. LongWriter-Zero (Wu et al., 2025b) trains its reward model based on coarse-grained attributes such as fluency, coherence, and helpfulness. In contrast, our method trains the reward model in a fine-grained manner by constructing training data according to the writing requirements in the instructions.

3 Method

As shown in Fig. 2, our method consists of two main components: (1) a fine-grained evaluation pipeline WEval for writing reward models that applies requirement dropout to construct a partial order and evaluates reward models by computing the correlation between their rankings of candidate responses and naturally formed golden rankings; and (2) a reinforcement learning training framework WRL that employs fine-grained Bradley–Terry training to writing reward models, which provide reward signals for policy model training via Group Relative Policy Optimization (Shao et al., 2024).

3.1 Evaluating Writing Reward Model

3.1.1 Seed Query Collection

To construct evaluation data for writing reward models, we collect seed instructions from large-scale user–LLM conversation datasets. Specifically, we draw from LMSYS-1M (Zheng et al., 2023), which contains around one million dialogues spanning 25 LLMs collected on the chat platform; WildChat (Zhao et al., 2024), comprising 652.1k conversations with GPT-3.5 and GPT-4; and PRISM (Kirk et al., 2024), a curated set of 8.0k dialogues designed to capture user preferences and behaviors.

Our evaluation dataset covers five main task types: (1) Creative Writing & Narrative, (2) Frameworks & Structured Plans, (3) Long-form Academic Writing, (4) Discussion & Expression Tasks, and (5) Informational & Practical Writing. Given the seed instructions and predefined task categories, we obtain task-aligned subsets through a filtering process. For each category, human annotators first select 10 high-quality prototype examples. Based on their instruction embeddings, we compute a category centroid. For a candidate prompt with embedding v_j, its relevance is measured by the cosine similarity:

$$\text{sim}(v_j, \text{centroid}) = \frac{v_j \cdot \text{centroid}}{\|v_j\| \, \|\text{centroid}\|}.$$

3.1.2 Query Variation

Building on the collected seed instructions, we introduce four major categories of writing requirements: (1) Content, which specifies the key information, themes, or points that the writing task should cover; (2) Style, which defines the tone, language, or stylistic manner expected in the response; (3) Format, which describes the organizational structure or presentation form of the writing task; and (4) Length, which sets constraints on the expected scope, such as the number of words, sentences, or paragraphs.

3.1.3 Requirement-Level Evaluation

Existing benchmarks provide coarse-grained, general evaluation of writing reward models but fail to assess performance on specific requirements in writing tasks. To address this gap, we propose a fine-grained evaluation pipeline WEval that evaluates reward models by comparing the correlation between their rankings of candidate responses and naturally formed golden rankings.

For each query q with n requirements $(c_1, c_2, \ldots, c_n)$, we generate n candidate responses.

# Inst.	# Requirements	Candidate Resp.	# Num
2750	13750	a_1	550
		a_2	550
		a_3	550
		a_4	550
		a_5	550

Table 1: Statistics of the evaluation dataset of WEval. # Inst. denotes the number of instructions. # Requirements denotes the total number of requirements. Candidate Resp. indicates the type of candidate responses generated for each query. # Num denotes the number of candidate responses of each type.

Specifically, we perform n rounds: in the first round, we use the original query with all n requirements; in the subsequent $n-1$ rounds, we randomly dropout 1, 2, …, $n-1$ requirements, respectively, and use DeepSeek-R1 (Guo et al., 2025) to generate an answer for each modified query. Since more requirement dropouts naturally result in worse requirement adherence, the responses naturally form an ordering where the degree of requirement adherence decreases as more requirements are dropped. This natural ordering serves as the golden ranking. During evaluation, the reward model ranks the n candidate responses $(a_1, a_2, \ldots, a_n)$ for each query q. We then assess the reward model's performance by comparing its predicted rankings with the golden rankings through correlation metrics, systematically evaluating the quality of writing reward models on specific requirements. Tab. 1 presents the evaluation dataset statistics.

Specifically, we use three metrics—**Correlation**, **IL**, and **PL**—to evaluate the ranking produced by the reward model. Let r_1 denote the ranking generated by the reward model and r_2 denote the golden ranking. The metrics are as follows:

Correlation measures the correlation between the two rankings:

$$\text{Correlation} = \frac{\sum_{1 \le i < j \le n} \operatorname{sgn}\big(r_1(i) - r_1(j)\big)\, \operatorname{sgn}\big(r_2(i) - r_2(j)\big)}{\frac{1}{2}\, n(n-1)}$$

IL (Instruction-Level) quantifies the proportion of items that occupy the same relative positions:

$$\text{IL}(r_1, r_2) = \frac{1}{n} \left| \{\, i \mid \text{pos}_{r_1}(i) = \text{pos}_{r_2}(i) \} \right|$$

PL (Prompt-Level) measures the proportion of cases where the two rankings are identical:

$$\text{PL}(r_1, r_2) = \begin{cases} 1, & \text{if } r_1 = r_2 \\ 0, & \text{otherwise} \end{cases}$$

3.2 WRL Training

3.2.1 Reward Model Training

As shown in Fig. 2, our training framework WRL constructs positive and negative examples by **applying dropout to the requirements** in writing queries, enabling fine-grained reward modeling. We train Qwen2.5-7B-Instruct as the writing reward model. Specifically, for each training sample, given a query q with requirements $(c_1, c_2, \ldots, c_n)$, the original response is treated as a chosen example a, while rejected examples a_{drop} are constructed by dropping requirements from q. The reward model is optimized using the Bradley–Terry loss (Bradley and Terry, 1952), encouraging higher scores for responses that better adhere to the requirements:

$$\mathcal{L}_w = -\log \sigma\big(r(q, a) - r(q, a_{\text{drop}})\big)$$

3.2.2 GRPO Training

We collect diverse instructions from WildChat-1M (Zhao et al., 2024) and filter out writing-related tasks using DeepSeek-R1, resulting in a total of 13,221 instructions. We use DeepSeek-R1 to extract all the requirements in the question for reward modeling. During GRPO training, given a query q, the policy model generates rollouts $(o_1, o_2, \ldots, o_n)$. For each rollout o_i, our trained reward model computes a reward $r(q, o_i)$. These rewards $(r_1, r_2, \ldots, r_n)$ are then used to update the policy model, creating a feedback loop for improving response generation.

4 Experiment

4.1 Experiment Setup

We evaluate our method on three categories of models: (1) **Proprietary LLMs**: o3-mini, Gemini-2.5-Pro-Preview, Claude-3-7-Sonnet, GPT-4o, and o1-Preview; (2) **Open-source LLMs**: DeepSeek-R1-0528, Qwen3 series (Yang et al., 2025), Qwen2.5 series, and Llama-3 series (Grattafiori et al., 2024); (3) **Writing-enhanced LLMs**: LongWriter-llama3.1-8B, LongWriter-glm4-9B (Bai et al., 2024), and LongWriter-Zero-32B (Wu et al., 2025b). We apply WRL to base models including Qwen2.5 series, Llama-3.1-8B-Instruct, Distill-Qwen-14B, and Qwen3-8B.

Models	Avg	Domains						Requirements					
		D1	D2	D3	D4	D5	D6	R1	C	R2	C	R3	C
Proprietary LLMs													
o3-2025-04-16	85.3	84.8	85.2	83.9	85.9	85.8	86.8	85.1	87.5	85.2	91.0	86.3	87.2
Gemini-2.5-pro-preview	83.1	83.2	81.5	83.0	84.5	84.5	82.1	83.6	86.5	83.9	90.5	83.4	84.0
Claude-3-7-sonnet	78.5	78.2	77.9	76.5	79.4	79.3	80.9	79.4	82.5	78.8	86.1	79.2	80.5
GPT-4o	75.5	74.4	73.4	74.4	77.9	75.9	78.1	76.8	81.6	75.8	85.5	76.1	76.7
o1-Preview	68.6	68.5	67.0	66.6	69.5	70.3	71.4	70.1	75.1	68.5	79.8	70.9	73.8
Open-source LLMs													
DeepSeek-R1-0528	83.2	83.2	81.5	81.6	85.7	84.1	84.4	84.2	87.3	83.7	89.4	83.8	82.7
Qwen3-235B-A22B-thinking	81.5	80.2	79.2	81.0	82.9	82.5	82.9	82.5	85.0	81.3	88.2	81.3	81.8
LongWriter-Zero-32B	80.3	80.7	80.3	80.2	76.1	83.6	81.0	79.9	83.4	80.8	86.8	80.2	82.1
Qwen3-235B-A22B	73.6	73.6	72.9	74.0	70.1	76.5	74.7	77.5	82.1	77.0	87.3	76.3	79.6
Qwen2.5-72B-instruct	65.3	65.8	63.4	63.8	62.8	68.1	67.9	65.8	70.5	65.9	78.7	66.4	68.0
LongWriter-glm4-9B	62.9	64.1	63.7	62.4	61.3	65.0	61.3	62.8	66.7	63.6	74.8	63.4	65.9
LongWriter-llama3.1-8B	58.0	60.1	59.3	57.6	56.0	58.4	56.7	58.1	61.4	58.6	67.6	59.1	63.0
Llama-3.3-70B-instruct	50.4	50.7	49.3	47.9	48.5	52.9	56.6	50.7	50.7	50.4	50.4	51.1	51.1
Capability-enhanced LLMs													
Qwen2.5-1.5B-Instruct	44.6	46.0	46.8	43.8	37.5	48.4	45.0	44.1	47.8	44.9	54.5	43.2	43.7
w/ WRL	**50.1**	**53.5**	**51.7**	**50.6**	**42.9**	**53.9**	**49.6**	**49.8**	**53.2**	**51.2**	**61.6**	**49.7**	**49.3**
Qwen2.5-7B-Instruct	57.0	58.6	55.9	55.3	53.0	59.5	60.1	57.4	62.3	57.7	70.2	56.6	55.6
w/ WRL	**64.4**	**66.6**	**63.7**	**64.9**	**59.0**	**69.1**	**65.7**	**65.1**	**68.6**	**65.8**	**78.3**	**64.8**	**67.3**
Llama-3.1-8B-Instruct	47.0	48.5	46.6	44.7	43.4	50.3	51.9	46.9	50.5	47.4	56.6	46.8	45.2
w/ WRL	**48.5**	**50.4**	**48.3**	**46.4**	**44.4**	**51.7**	**52.7**	**48.2**	**51.7**	**49.0**	**59.0**	**48.5**	**47.9**
Distill-Qwen-14B	60.7	60.0	59.7	61.7	58.0	62.5	63.9	62.1	67.4	61.7	73.1	60.7	61.2
w/ WRL	**66.7**	**67.2**	**64.8**	**67.2**	**64.3**	**69.0**	**69.8**	**68.3**	**72.7**	**67.8**	**80.8**	**67.5**	**71.0**
Qwen3-8B	74.9	74.9	73.9	75.0	73.5	76.8	76.5	75.6	80.1	75.7	84.8	74.6	78.1
w/ WRL	**76.2**	**76.3**	**74.6**	**76.5**	**75.2**	**78.3**	**78.0**	**77.6**	**80.7**	**76.9**	**86.6**	**77.0**	**81.3**

Table 2: Performance of different LLMs on WritingBench across six domains and three writing requirements. Scores are normalized from a 0–10 range to a 100-point scale. The domains include: (D1) Academic & Engineering, (D2) Finance & Business, (D3) Politics & Law, (D4) Literature & Art, (D5) Education, and (D6) Advertising & Marketing. The writing requirements assessed are: (R1) Style, (R2) Format, and (R3) Length. Here, "C" indicates category-specific scores. The latest results are available on the online leaderboard.

We evaluate writing models on multiple benchmarks to assess both in-domain and out-of-domain performance. For **in-domain evaluation**, we use WritingBench (Wu et al., 2025d), LongWriter (Bai et al., 2024), and Arena-Write (Wu et al., 2025b). For **out-of-domain evaluation**, we test on DeepResearch Bench-RACE (Du et al., 2025) and FINDER_DEFT (Zhang et al., 2025). For **reward model evaluation**, we evaluate on our constructed evaluation dataset.

4.2 Performance of Writing Models

As reported in Tab. 2, WRL consistently improves the performance of base models across different parameter scales on WritingBench. For example, Qwen2.5-7B-Instruct improves by 7.4 and outperforms writing-oriented models such as LongWriter-llama3.1-8B and LongWriter-glm4-9B with comparable model sizes. Qwen2.5-1.5B-Instruct improves by 5.5, while Distill-Qwen-14B and Qwen3-8B achieve gains of 6.0 and 1.3, respectively. Notably, Qwen3-8B enhanced with WRL even surpasses proprietary models such as GPT-4o and o1-Preview. These gains are observed across all six domains and three writing requirements, suggesting that our fine-grained reward modeling strategy generalizes well across different writing scenarios.

Tab. 3 further shows that, after training with our method, the model achieves markedly better writing quality across multiple evaluation dimensions, including Relevance, Accuracy, Coherence, Clarity, Breadth and Depth, and Reading Experience, demonstrating the practical value of our method in real-world writing applications.

Similar trends can be observed in Fig. 3, where models trained with WRL achieve higher win rates against strong baselines on Arena-Write. Qwen2.5-7B-Instruct-WRL improves the win rate by 4.6%, and Distill-Qwen-14B-WRL achieves the largest gain of 8.7%, further indicating better alignment with human preferences.

Model	Relevance	Accuracy	Coherence	Clarity	Breadth and Depth	Reading Experience	Total
LongWriter-glm4-9B	95.0	90.0	91.7	88.5	73.3	83.5	87.0
LongWriter-llama3.1-8B	91.5	88.3	86.0	85.6	66.7	76.5	82.4
Qwen2.5-1.5B-Instruct	68.5	71.0	65.2	65.0	42.1	55.0	61.1
Qwen2.5-1.5B-Instruct-WRL	**76.9**	**77.7**	**70.2**	**70.0**	**50.4**	**57.5**	**67.1**
Llama-3.1-8B-Instruct	89.8	81.9	80.6	79.2	53.3	66.9	75.3
Llama-3.1-8B-Instruct-WRL	**92.3**	**84.0**	**81.9**	**79.8**	**55.2**	**69.8**	**77.2**
Qwen2.5-7B-Instruct	91.7	89.6	80.8	80.6	67.7	74.6	80.8
Qwen2.5-7B-Instruct-WRL	**92.9**	**90.0**	**82.7**	**82.9**	**72.3**	**76.5**	**82.9**
Distill-Qwen-14B	97.1	**92.9**	**94.4**	**92.1**	70.0	84.0	88.4
Distill-Qwen-14B-WRL	**97.5**	91.9	92.7	90.6	**76.3**	**85.2**	**89.0**
Qwen3-8B	**96.2**	93.3	90.2	89.6	79.2	**86.3**	89.1
Qwen3-8B-WRL	95.6	**93.5**	**91.3**	**89.8**	**81.9**	85.4	**89.6**

Table 3: Performance of different models on LongWriter. The results show that our method can significantly improve the writing quality of the model.

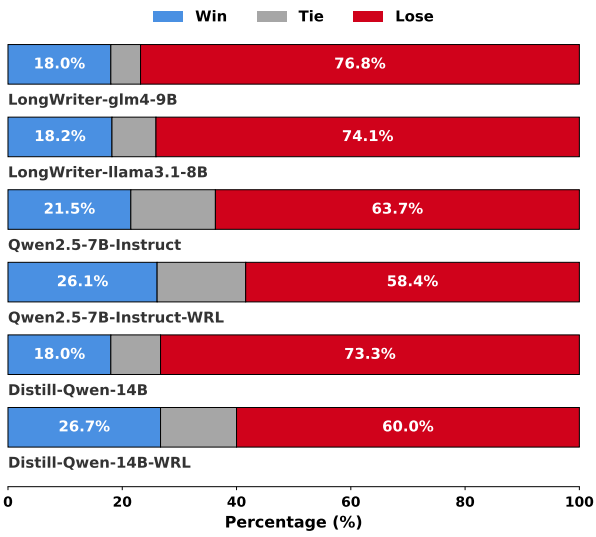

Figure 3: Performance of different models on Arena-Write against six strong baselines.

Models	Correlation	IL	PL
Qwen2.5-7B-Instruct-as-a-judge	85.4	92.9	60.9
Qwen2.5-72B-Instruct-as-a-judge	93.6	96.8	**78.7**
Writing-Critic-7B	80.6	90.8	42.7
Skywork-Reward-V2-Llama-3.1-8B	75.4	88.7	32.9
Skywork-Reward-V2-Qwen3-8B	82.5	91.7	45.3
Our-RM-7B	**94.6**	**97.3**	78.0
Human	95.7	97.9	79.4

Table 4: Performance of different reward models on the evaluation dataset. Our trained reward model demonstrates superior performance in fine-grained requirement adherence evaluation.

4.3 Performance of Writing Reward Models

As shown in Tab. 4, we evaluate a diverse set of reward models on our evaluation dataset, including both LLM-as-a-judge methods and trained reward models. Compared with strong baselines, our 7B reward model achieves the best performance consistently across all three metrics. Specifically, it obtains 94.6 on Correlation, 97.3 on IL, and 78.0 on PL, outperforming all competing reward models by clear margins. These results demonstrate that our reward model is more aligned with fine-grained writing quality assessment and provides more reliable reward signals than both judge-style prompting methods and existing reward modeling baselines. In addition, the rankings of human annotators achieve strong performance on all three metrics, further validating that our evaluation protocol is highly consistent with human judgments.

4.4 Ablation Studies

As shown in Tab. 5, we study the impact of different training strategies and reward models on downstream performance. Our WRL method achieves the best results on both WritingBench and Arena-Write, consistently outperforming all other approaches, including supervised fine-tuning, LLM-as-a-judge RL, and WRL with existing reward models. Notably, replacing the reward model with Our-RM-7B further boosts performance to 64.4 on WritingBench and 26.05 on Arena-Write, yielding the strongest overall results. Moreover, the relative ranking of reward models in downstream RL is consistent with their evaluation results in Tab. 4. These results further validate the effectiveness of our reward model evaluation protocol and show that it is highly predictive of reward model utility in practical RL settings.

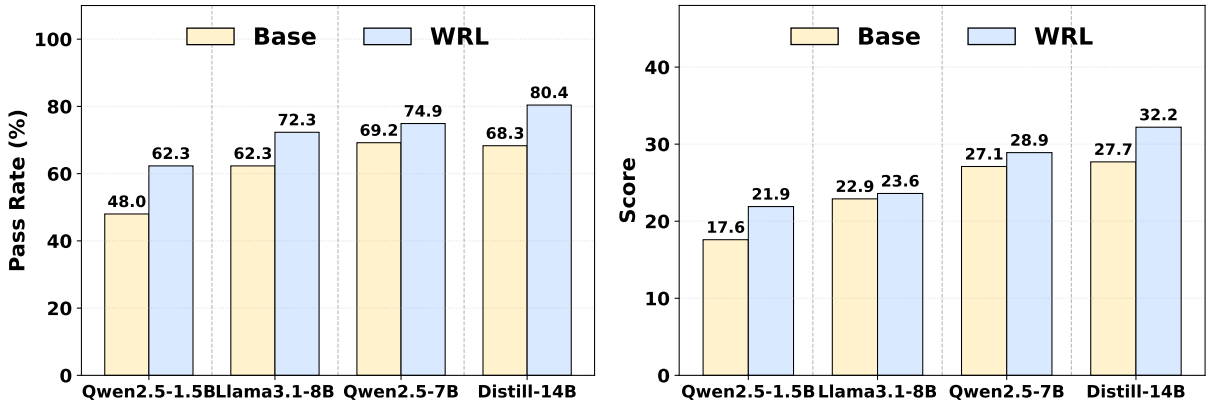

Figure 4: Performance of different models on DeepResearch-related benchmarks, including FINDER_DEFT (left) and DeepResearch Bench-RACE (right).

Models	Reward Models	WritingBench	Arena-Write
Qwen-2.5-7B-Inst.	-	57.0	21.51
w/ SFT	-	59.4	16.47
w/ llm-as-Judge RL	Qwen2.5-7B-Inst.	62.1	24.70
w/ WRL	Skywork-Qwen3	61.2	23.53
w/ WRL	Our-RM-7B	**64.4**	**26.05**

Table 5: Ablation study on different training methods and reward models across WritingBench and Arena-Write benchmarks.

Models	Correlation	IL	PL
Qwen2.5-7B-Instruct-as-a-judge	83.5	92.0	59.6
Writing-Critic-7B	80.6	90.8	43.5
Skywork-Reward-V2-Llama-3.1-8B	82.8	91.9	44.0
Our-RM-7B	**92.5**	**96.5**	**76.0**

Table 6: Performance of different reward models on the evaluation dataset constructed by GPT-4o.

Models	Correlation	IL	PL
Qwen2.5-7B-Instruct-as-a-judge	84.2	92.4	59.6
Writing-Critic-7B	80.6	90.8	41.5
Skywork-Reward-V2-Llama-3.1-8B	79.5	90.6	43.5
Our-RM-7B	**91.2**	**95.8**	**68.4**

Table 7: Performance of different reward models on the evaluation dataset constructed by Gemini-2.5-Pro.

4.5 Generalizability

Our WRL method generalizes well to DeepResearch tasks and consistently improves the quality of generated research reports. As shown in Fig. 4, WRL yields stable gains on both DeepResearch Bench-RACE and FINDER_DEFT across models of different scales. On DeepResearch Bench-RACE, Qwen2.5-1.5B-Instruct-WRL improves the overall score by 4.3, while Distill-Qwen-14B-WRL achieves the largest gain of 4.5. On FINDER_DEFT, WRL also substantially improves checklist pass rate, with gains of 14.3 for Qwen2.5-1.5B-Instruct, 12.1 for Distill-Qwen-14B, and 10.0 for Llama-3.1-8B-Instruct. These results show that WRL transfers effectively to out-of-domain research report generation tasks.

To further examine the generalization of our evaluation framework WEval, we reconstruct the evaluation datasets using two different teacher models, GPT-4o and Gemini-2.5-Pro, and report the results in Tab. 6 and Tab. 7, respectively. We observe that the overall ranking of reward models remains highly consistent with that in Tab. 4 across both settings. In particular, Our-RM-7B consistently achieves the best performance on all three metrics, while Qwen2.5-7B-Instruct-as-a-judge remains the strongest baseline among the compared methods. This consistency across datasets constructed by different models indicates that our evaluation results are robust to the choice of data construction model. These findings further demonstrate the strong generalization ability of our evaluation method and validate its reliability for reward model assessment.

We further verify the generalization of our reward model training approach on Qwen3-32B-Base by comparing with Writing-Zero (Jia et al., 2025) on RewardBench (Lambert et al., 2025). Writing-Zero trains a reward model through a multi-stage pipeline involving preference filtering, cold-start supervised fine-tuning with reasoning traces, and GRPO-based reinforcement learning. In contrast, our method directly constructs preference pairs from dropped requirements and the natural partial order implied by instructions, and optimizes the reward model with the Bradley–Terry loss. As shown in Tab. 8, our trained model substantially improves

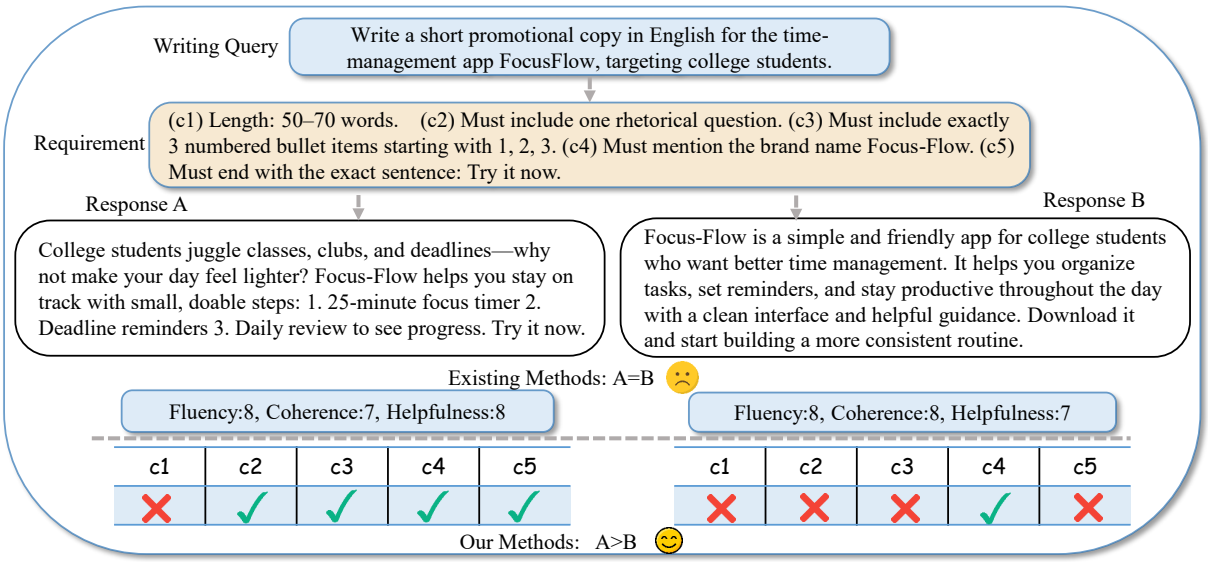

Figure 5: Case study between previous writing task evaluation paradigms and our method. Our approach provides fine-grained, requirement-level evaluation and reward modeling.

Model	Chat	Chat Hard	Safety	Reasoning	Avg
Qwen-32B-Base	93.9	67.3	87.0	84.3	83.1
Claude-3.5-Sonnet	96.4	74.0	81.6	84.7	84.2
Pairwise GenRM*	–	–	–	–	**87.4**
Our Trained RM	97.8	64.9	85.1	93.4	<u>85.3</u>

Table 8: Comparison of different reward models on RewardBench. * denotes that the results are taken directly from the original paper Writing-Zero. We use **bold** for the best results and <u>underlined</u> for the second-best results. Our reward model training method is simple yet effective.

over the Qwen3-32B-Base baseline and achieves strong overall performance, reaching 97.8 on Chat, 85.1 on Safety, 93.4 on Reasoning, and 85.3 on average. Although it slightly underperforms Pairwise GenRM on Chat Hard, it approaches the performance of Writing-Zero's model with a much simpler training pipeline. These results further demonstrate that our method generalizes effectively to larger base models and provides a simple yet strong solution for reward model training.

4.6 Analysis

Fig. 5 presents a representative case study comparing existing writing evaluation paradigms with our method. Although previous methods may regard Responses A and B as comparable under coarse-grained criteria such as fluency, coherence, and helpfulness, this assessment fails to capture whether the responses truly satisfy the detailed task requirements. In this example, Response A violates the length constraint, while Response B misses multiple explicit requirements, including the rhetorical question, the numbered bullet format, the length requirement, and the required ending sentence. Our method explicitly decomposes the writing task into fine-grained requirement-level criteria and evaluates each response against these concrete requirements. As a result, it can accurately identify that Response A satisfies more key requirements than Response B, leading to a more reliable preference judgment. This case demonstrates that, compared with conventional holistic evaluation paradigms, our method provides a more fine-grained, requirement-aware, and accurate assessment for complex writing tasks.

5 Conclusion

In this paper, we first propose a fine-grained evaluation pipeline WEval for writing reward models that assesses their performance on specific writing requirements through correlation metrics with naturally formed golden rankings constructed via requirement dropout. We introduce a reinforcement learning training framework, WRL, that employs fine-grained Bradley–Terry training to train writing reward models, which are then used to provide reward signals for RL training. Experimental results demonstrate that our trained writing reward models effectively improve policy models' writing capabilities, and the consistency between reward model evaluation results and downstream RL training performance validates the effectiveness of our

evaluation approach.

Limitations

Our study has several limitations. First, we do not evaluate our training method on larger-scale models (e.g., 32B parameters or above). Nevertheless, our approach is model-agnostic and shows strong generalization across different architectures and training settings. Second, the constructed evaluation datasets cover only a limited range of requirement types. Although they already capture several representative writing constraints, they do not yet fully reflect the diversity and complexity of real-world writing tasks.

Ethical Considerations

We discuss the potential ethical concerns as follows. The annotation of instruction task categories and the ranking of the reward model evaluation dataset were conducted by three annotators with computer science backgrounds recruited by our institution. The annotation team remained anonymous to the authors. We ensured that the privacy rights of all annotators were respected throughout the annotation process. All annotators were compensated above the local minimum wage and gave informed consent for the data to be used for research purposes. Disagreements in annotation were resolved through majority voting. The annotation details are shown in Appx. A.8.

Acknowledgments

We acknowledge the use of Cursor as an AI-assisted writing tool during the preparation of this manuscript. Its role was to polish the language of the initial draft. All core ideas presented in the paper were conceived independently by the authors.

References

Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. Longwriter: Unleashing 10,000+ word generation from long context llms. *arXiv preprint arXiv:2408.07055*.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Jianghao Chen, Wei Sun, Qixiang Yin, Zhixing Tan, and Jiajun Zhang. 2025. Ace-rl: Adaptive constraint-enhanced reward for long-form generation reinforcement learning. *arXiv preprint arXiv:2509.04903*.

Fengxiang Cheng. 2026. Empowering llms with symbolic representation and reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 41040–41041.

Yujuan Ding, Yunshan Ma, Wenqi Fan, Yige Yao, Tat-Seng Chua, and Qing Li. 2024. Fashionregen: Llm-empowered fashion report generation. In *Companion Proceedings of the ACM Web Conference 2024*, pages 991–994.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. 2025. Deepresearch bench: A comprehensive benchmark for deep research agents. *arXiv preprint arXiv:2506.11763*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Qianyu He, Jie Zeng, Qianxi He, Jiaqing Liang, and Yanghua Xiao. 2024. From complex to simple: Enhancing multi-constraint complex instruction following ability of large language models. *arXiv preprint arXiv:2404.15846*.

Hui Huang, Yancheng He, Wei Liu, Muyun Yang, Jiaheng Liu, Kehai Chen, Bing Xu, Conghui Zhu, Hailong Cao, and Tiejun Zhao. 2026. Long-form rewardbench: Evaluating reward models for long-form generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 31149–31157.

Ruipeng Jia, Yunyi Yang, Yongbo Gai, Kai Luo, Shihao Huang, Jianhe Lin, Xiaoxi Jiang, and Guanjun Jiang. 2025. Writing-zero: Bridge the gap between non-verifiable tasks and verifiable rewards. *arXiv preprint arXiv:2506.00103*.

Hannah R Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, et al. 2024. The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *Advances in Neural Information Processing Systems*, 37:105236–105344.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. 2025. Rewardbench: Evaluating reward models for language modeling. In *Findings*

of the Association for Computational Linguistics: NAACL 2025, pages 1755–1797.

Weiyue Li, Mingxiao Song, Zhenda Shen, Dachuan Zhao, Yunfan Long, Yi Li, Yongce Li, Ruyi Yang, and Mengyu Wang. 2026. Llm review: Enhancing creative writing via blind peer review feedback. *arXiv preprint arXiv:2601.08003*.

Jianxing Liao, Tian Zhang, Xiao Feng, Yusong Zhang, Rui Yang, Haorui Wang, Bosi Wen, Ziying Wang, and Runzhi Shi. 2025. Rlmr: Reinforcement learning with mixed rewards for creative writing. *arXiv preprint arXiv:2508.18642*.

David Y Liu, Xanthe Muston, Aditya Joshi, and Sebastian Sequoiah-Grayson. 2026. Retell, reward, repeat: Reinforcement learning for narrative theory-informed story generation. *arXiv preprint arXiv:2601.17226*.

Tohru Nagano, Gakuto Kurata, Samuel Thomas, Hong-Kwang J Kuo, Daniel Bolanos, Hyun Jung, and George Saon. 2025. Llm based text generation for improved low-resource speech recognition models. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Hao Peng, Yunjia Qi, Xiaozhi Wang, Bin Xu, Lei Hou, and Juanzi Li. 2025. Verif: Verification engineering for reinforcement learning in instruction following. *arXiv preprint arXiv:2506.09942*.

Chau Minh Pham, Simeng Sun, and Mohit Iyyer. 2024. Suri: Multi-constraint instruction following for long-form text generation. *arXiv preprint arXiv:2406.19371*.

Shanghaoran Quan, Tianyi Tang, Bowen Yu, An Yang, Dayiheng Liu, Bofei Gao, Jianhong Tu, Yichang Zhang, Jingren Zhou, and Junyang Lin. 2024. Language models can self-lengthen to generate long texts. *arXiv preprint arXiv:2410.23933*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Qingyu Ren, Qianyu He, Bowei Zhang, Jie Zeng, Jiaqing Liang, Yanghua Xiao, Weikang Zhou, Zeye Sun, and Fei Yu. 2025a. Instructions are all you need: Self-supervised reinforcement learning for instruction following. *arXiv preprint arXiv:2510.14420*.

Qingyu Ren, Jie Zeng, Qianyu He, Jiaqing Liang, Yanghua Xiao, Weikang Zhou, Zeye Sun, and Fei Yu. 2025b. Step-by-step mastery: Enhancing soft constraint following ability of large language models. *arXiv preprint arXiv:2501.04945*.

Jaydip Sen, Rohit Pandey, and Hetvi Waghela. 2025. Context-enhanced contrastive search for improved llm text generation. *arXiv preprint arXiv:2504.21020*.

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*.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. 2024. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*.

Saranya Venkatraman, Nafis Irtiza Tripto, and Dongwon Lee. 2025. Collabstory: Multi-llm collaborative story generation and authorship analysis. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3665–3679.

Phoebe J Wang and Max Kreminski. 2024. Guiding and diversifying llm-based story generation via answer set programming. *arXiv preprint arXiv:2406.00554*.

Zhuhao Wang, Yihua Sun, Zihan Li, Xuan Yang, Fang Chen, and Hongen Liao. 2025. Llm-rg4: Flexible and factual radiology report generation across diverse input contexts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8250–8258.

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 17171–17197.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. 2025a. A survey on llm-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, 51(1):275–338.

Yuhao Wu, Yushi Bai, Zhiqiang Hu, Roy Ka-Wei Lee, and Juanzi Li. 2025b. Longwriter-zero: Mastering ultra-long text generation via reinforcement learning. *arXiv preprint arXiv:2506.18841*.

Yuhao Wu, Yushi Bai, Zhiqiang Hu, Juanzi Li, and Roy Ka-Wei Lee. 2025c. Superwriter: Reflection-driven long-form generation with large language models. *arXiv preprint arXiv:2506.04180*.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, et al. 2025d. Writingbench: A comprehensive benchmark for generative writing. *arXiv preprint arXiv:2503.05244*.

Lei Xu, Pierre Beckmann, Marco Valentino, and André Freitas. 2026. Adaptive llm-symbolic reasoning via dynamic logical solver composition. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1187–1208.

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*.

Dong Yuan, Eti Rastogi, Gautam Naik, Sree Prasanna Rajagopal, Sagar Goyal, Fen Zhao, Bharath Chintagunta, and Jeffrey Ward. 2024a. A continued pretrained llm approach for automatic medical note generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 565–571.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E Weston. 2024b. Self-rewarding language models. In *Forty-first International Conference on Machine Learning*.

Dingling Zhang, He Zhu, Jincheng Ren, Kangqi Song, Xinran Zhou, Boyu Feng, Shudong Liu, Jiabin Luo, Weihao Xie, Zhaohui Wang, et al. 2025. How far are we from genuinely useful deep research agents? *arXiv preprint arXiv:2512.01948*.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P Xing, et al. 2023. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv preprint arXiv:2309.11998*.

A Appendix

A.1 Prompt for Evaluation Dataset Construction

To construct our evaluation dataset, we employ a systematic prompt engineering approach that generates diverse and challenging writing tasks. As shown in Tab. 9, we use a comprehensive prompt template that guides the generation of atomic constraints to enhance the difficulty of seed questions. The prompt consists of two main components: (1) **Task Description**, which specifies the requirements for designing five new atomic constraints that significantly increase the difficulty of answering a given seed question, and (2) **Constraint References**, which provides eight categories of constraint types including length, format, style and content constraints. Each category contains multiple sub-types with specific examples, enabling the generation of diverse and granular constraints that cover various aspects of writing tasks. The constraints must be specific, actionable, and non-conflicting, ensuring that the resulting evaluation dataset contains high-quality and challenging writing instructions.

A.2 Details of Tasks in the Evaluation Dataset

1. Creative Writing & Narrative

Scope: Stories, songs, and scripts emphasizing imagination, narrative diversity, and emotional resonance. **Purpose**: Engage readers through plot, character development, dialogue, and creative expression. **Techniques**: Metaphors, similes, plot twists, rhythm, rhyme, suspense, and perspective shifts. **Audience**: Children, young adults, or mature audiences. **Formats**: Short stories, musicals, film scripts, serialized narratives, lyrical compositions.

2. Frameworks & Structured Plans

Scope: Outlines, conceptual frameworks, structured workflows, and planning templates. **Purpose**: Organize ideas, guide projects, and present logical sequences for execution. **Techniques**: Hierarchical structures, bullet points, flowcharts, stepwise reasoning, and modular design. **Audience**: Teams, students, or professionals requiring clarity and actionable plans. **Formats**: Project plans, design frameworks, step-by-step guides, and workflow diagrams.

3. Long-form Academic Writing

Scope: Extended essays, research papers, technical reports, and scholarly articles integrating citations and empirical data. **Purpose**: Demonstrate analysis, argumentation, and evidence-based reasoning. **Techniques**: Formal tone, structured headings, literature reviews, methodology sections, and in-text citations. **Audience**: Academics, researchers, and specialists in a given discipline. **Formats**: Thesis, journal articles, white papers, systematic reviews, and technical manuals.

4. Discussion & Expression Tasks

Scope: Question–answer formats, interviews, debates, or dialogue-driven prompts emphasizing reasoning and perspective exchange. **Purpose**: Facilitate critical thinking, reflective responses, and conversational clarity. **Techniques**: Probing questions, counterarguments, structured dialogues, role-playing, and argument scaffolding. **Audience**: Classroom settings, workshops, research interviews, or collaborative problem solving. **Formats**: Interviews, Socratic dialogues, debate transcripts, discussion boards, reflective essays.

5. Informational & Practical Writing

Scope: Functional, utilitarian writing such as reports, letters, instructions, and procedural documentation. **Purpose**: Communicate facts, procedures, or practical guidance clearly and efficiently. **Techniques**: Concise language, structured formatting, headings, numbered steps, tables, and bullet points. **Audience**: General public, business professionals, students, or technical users. **Formats**: Business reports, user manuals, formal letters, policy documents, and how-to guides.

A.3 Details of Requirements in the Evaluation Dataset

1. Length Requirements

Word/Sentence/Paragraph Limits: Total word count (e.g., 120–150 words), exact sentence count (e.g., 5), paragraph limits (e.g., 3 paragraphs, each $\leq$40 words). **Proportions and Lengths**: Proportional allocation (e.g., conclusion 30%), character restrictions ($\leq$500 including spaces), sentence length limits (e.g., each sentence $<$15 words or $\geq$10 words).

[Task Description]
1. You will receive a [Seed Question]. Your task is to design five new atomic constraints that significantly increase the difficulty of answering the question.
2. These constraints must be added without modifying the original [Seed Question].
3. Constraints must be specific and granular, avoiding vague instructions (e.g., "brief", "formal"), and must not conflict with each other.
4. The five constraints should cover multiple types and cannot all belong to the same type.
5. Each constraint must include clear, actionable details such as specific numbers, scenarios, times, entities, sequences, or language structures.
6. Encourage the inclusion of reasoning, metaphor, classification, ordering, or background elaboration to increase difficulty.
7. The output must be in strict JSON format with keys c1, t1, c2, t2, c3, t3, c4, t4, c5, t5 representing five constraints and their types.
[Constraint References]
1. Length Constraints
Word Limit: Set a word count range, e.g., "Use 120-150 words."
Sentence Limit: Restrict sentences, e.g., "Write exactly 5 sentences."
Paragraph Limit: Limit paragraphs and length, e.g., "Use 3 paragraphs, each at most 40 words."
Proportional Distribution: Allocate proportions, e.g., "Conclusion is 30% of text."
Character Limit: Restrict characters, e.g., "at most 500 characters, including spaces."
Sentence Length: Limit sentence length, e.g., "Each sentence less than 15 words" or "at least 10 English words."
2. Format Constraints
Customized formatting for specific needs, e.g., "Summarize main points in an unordered list."
Formatting standards for specialized applications, e.g., "Conform to electronic medical record format."
Defines how to highlight or emphasize parts of the text using styles or symbols, e.g., "Bold all key terms and use warning symbols before warnings."
3. Style Constraints
Tone: Adopt a tone, e.g., "Humorous and sarcastic."
Rhetorical Devices: Include devices, e.g., "Use at least 2 metaphors."
Audience: Target audience, e.g., "Explain for elementary students."
Identity/Voice: Write from perspective, e.g., "As a historian" or "Retired elder."
Emotional Appeal: Use emotional tone, e.g., "Evoke empathy" or "Create urgency."
Literary Device: Include techniques, e.g., "Use 1 parallelism, 1 rhetorical question."
Cultural Voice: Adopt cultural style, e.g., "Tang dynasty poet" or "Japanese haiku style."
4. Content Constraints
Required Entities: Include entities, e.g., "Name 3 scientists."
Chronological Order: Present sequentially, e.g., "Events in chronological order."
Data Requirement: Include data, e.g., "Use at least 2 statistical data points."
Thematic Coverage: Cover themes, e.g., "Economic, environmental, social aspects."
Time-Space Perspective: Use unique perspective, e.g., "Song dynasty" or "Arctic explorer."
Interdisciplinary Focus: Combine disciplines, e.g., "Physics and philosophy."
Counterargument: Address opposition, e.g., "Discuss 1 counterargument, rebut with evidence."
Hypothetical Scenario: Use hypotheticals, e.g., "Write from lunar base about future tech."
[Seed Question]
{raw_question}

Table 9: Complete prompt for constructing evaluation datasets

2. Format Requirements

Text Structure: Custom paragraphs, headings, emphasis, examples, bullet points. **Professional Standards**: Specific application formats (e.g., electronic medical records). **Emphasis Rules**: Text highlighting methods (e.g., bold key terms, use "!" for warnings).

3. Style Requirements

Tone and Voice: Humorous, formal, or perspective-based (e.g., historian, retired elder). **Rhetorical Devices**: Metaphors, parallelism, rhetorical questions, etc. **Audience and Cultural Style**: Adjust for audience, culture, or era (e.g., elementary students, Tang dynasty poet). **Emotional Appeal**: Evoke empathy, urgency, or other emotions.

4. Content Requirements

Entities and Data: Include specific entities (e.g., scientists) and data points. **Chronology and Themes**: Sequential events and multiple aspects (economic, environmental, social). **Perspective and Interdisciplinary Focus**: Unique time-space viewpoints or combined disciplines. **Counterarguments and Hypotheticals**: Address opposing views or create hypothetical scenarios.

A.4 Evaluation Benchmarks

A.4.1 In-Domain Benchmarks:

WritingBench (Wu et al., 2025d): A comprehensive benchmark comprising 1,239 tasks designed to evaluate LLMs across 6 core writing domains and 100 subdomains, encompassing creative, persuasive, informative, and technical writing. WritingBench supports evaluation of multiple requirement types including length, format, and style constraints, making it suitable for assessing models' ability to follow diverse writing instructions.

LongWriter (Bai et al., 2024): A benchmark suite consisting of two components: LongBench-Write and LongWrite-Ruler. LongBench-Write focuses on measuring long-form output quality across multiple dimensions including relevance, accuracy, coherence, clarity, breadth and depth, and reading experience, as well as output length. LongWrite-Ruler is designed as a lightweight stress test to evaluate the model's maximum output length capability, measuring how many words a model can generate in a single response.

Arena-Write (Wu et al., 2025b): A small-scale benchmark of 100 user writing tasks collected from real-world scenarios, designed to evaluate long-form generation models in realistic settings. Each task covers diverse formats such as social posts, essays, and reports, with many requiring outputs over 2,000 words. The benchmark uses human preference judgments to assess model performance through win rate comparisons against strong baselines.

A.4.2 Out-of-Domain Benchmarks:

DeepResearch Bench-RACE (Du et al., 2025): A comprehensive benchmark for evaluating deep research agents' ability to generate high-quality research reports. The benchmark assesses multiple dimensions including comprehensiveness (coverage of research topics), depth (level of detail and analysis), instruction following (adherence to specific requirements), readability (clarity and organization), and overall quality. It tests models' generalization capabilities to research-oriented writing tasks that require extensive knowledge integration and structured output.

FINDER_DEFT (Zhang et al., 2025): A benchmark designed to evaluate models' ability to follow detailed formatting and content requirements in research report generation. The benchmark uses a checklist-based evaluation approach, measuring the pass rate of generated reports against a comprehensive set of formatting and content criteria. It focuses on assessing models' precision in adhering to specific structural and content requirements, making it particularly suitable for evaluating fine-grained instruction following capabilities.

A.5 Baselines

LongWriter-llama3.1-8B (Bai et al., 2024): A writing-specialized model based on Meta-Llama-3.1-8B, trained with SFT and DPO. Capable of generating 10,000+ words.

LongWriter-glm4-9B (Bai et al., 2024): A writing-enhanced model based on GLM-4-9B, trained with SFT and DPO.

LongWriter-Zero-32B (Wu et al., 2025b): A purely RL-based model trained with coarse-grained reward attributes (fluency, coherence, helpfulness).

A.6 Evaluation Prompt Arena-Write

As shown in Tab. 10, we provide the prompts used for the Arena-Write evaluation.

Act as an impartial judge and evaluate the quality of the written responses provided by two AI assistants to the user's writing prompt below.
You will be given Assistant A's response and Assistant B's response. Your job is to determine which assistant's writing is superior.
Evaluation Criteria:
1. Relevance and Completeness:
- Does the assistant fully respond to the writing prompt?
- Does the length meet the user's query expectations?
- Is the content relevant to the topic?
- Does it provide sufficient depth, length, and detail, rather than drifting off-topic or being simplistic?
2. Writing Quality:
- Evaluate whether the assistant's writing is clear, fluent, and free of obvious grammatical errors.
- The overall quality of the writing should be high, with elegance.
3. Creativity and Originality:
- If applicable, assess the creativity of the response.
- Does the assistant offer fresh perspectives, unique insights, or demonstrate a certain level of originality?
4. Specificity and Detail:
- Determine whether the assistant provides concrete examples or detailed explanations.
- Properly justified repetition is permissible.
5. Tone and Style:
- Is the tone appropriate for the writing prompt?
- Is the writing style consistent throughout?
- Consider whether it aligns with the expectations of the intended audience or writing purpose.
After evaluating each response based on these factors, determine which one is superior, provide an explanation, and then select one of the following final verdicts:
- **Assistant A is significantly better:** [[A»B]]
- **Assistant A is slightly better:** [[A>B]]
- **Tie, relatively the same:** [[A=B]]
- **Assistant B is slightly better:** [[B>A]]
- **Assistant B is significantly better:** [[B»A]]
Example output: My final verdict is tie: [[A=B]].

Table 10: Prompts used for the Arena-Write evaluation (Wu et al., 2025b).

A.7 RL Training Implementation Details

We conduct training using the VeRL framework with the GRPO algorithm on the subset of *WildChat* dataset. Prompts and responses are truncated to a maximum length of 12000 tokens each. Data are shuffled with a fixed random seed of 1. Rollouts are generated with a rollout batch size of 384. Training is performed on a single node with 8 H200 GPUs. The global batch size is set to 96, with micro-batches of size 2 per device for policy updates and micro-batches of size 8 per device for experience generation. Rollouts are sampled with a temperature of 1.0 and a group size of 5, using tensor parallelism of size 2 and a maximum of 25,000 batched tokens.

A.8 Annotation Details

Annotators were asked to complete two annotation tasks. We present the *Instructions Given To Participants* as follows:

Task 1: Instruction category annotation. For each writing instruction, assign it to the single most appropriate task category based on its main purpose and form. Use the following five categories defined in the paper: (1) Creative Writing & Narrative, (2) Frameworks & Structured Plans, (3) Long-form Academic Writing, (4) Discussion & Expression Tasks, and (5) Informational & Practical Writing. Choose the category that best reflects the primary writing objective of the instruction.

Task 2: Response ranking based on requirement adherence. For each query, you will be shown multiple candidate responses. Rank these responses according to how well they satisfy the requirements in the original writing prompt. A response should receive a higher rank if it follows more of the required constraints. You should prioritize requirement satisfaction rather than relying only on general impressions such as fluency, coherence, or helpfulness. Finally, provide a ranking of the candidate responses from best to worst.