ChapterRL: Chapter-Aware Reward Framework for Novel Generation Reinforcement Learning

Anonymous ACL submission

Abstract

Multi-chapter novel generation requires models to balance two quality dimensions simultaneously: chapter-level reader engagement and global narrative rhythm. Existing approaches have mainly focused on planning, memory, and execution, while paying limited attention to the mismatch between training signals and reader experience. We propose **ChapterRL**, which reformulates multi-chapter generation as a reward design problem. Specifically, ChapterRL introduces two complementary reward signals. **Reading Depth Residual** (RDR) derives debiased chapter-level quality signals from reader continuation and completion behavior through genre–position stratified normalization. **Dynamic Rhythm Trajectory** (DRT) uses an arousal estimator to impose a threshold-gated penalty on systematic rhythm collapse in the later part of a story. The two signals correspond to chapter attractiveness and global narrative momentum, and are jointly integrated into reinforcement learning training. We evaluate the full ChapterRL policy on three automatic benchmarks and separately validate its RDR reward model on task-aligned narrative domains of the Chinese WritingPreferenceBench. Results show that ChapterRL improves chapter quality and rhythm stability over supervised fine-tuning and single-reward variants, while RDR exhibits strong narrative preference alignment as an efficient scalar reward model.Code is available at anonymous/ChapterRL.

1 Introduction

Multi-chapter novel generation lies at the intersection of long-form language modeling and narrative understanding. Unlike single-turn creative writing, it requires a model to maintain two indispensable forms of quality over thousands of tokens and across multiple chapters: **chapter-level engagement**, where each chapter must be sufficiently compelling to motivate continued reading, and **global narrative rhythm**, where the entire story must sustain coherent emotional tension, pacing, and forward momentum. Recent progress includes improved outline control (Yang et al., 2023; Chung et al., 2022), pacing-oriented planning (Wang et al., 2023a), retrieval-augmented generation (Wen et al., 2023), and hierarchical systems with memory and critique (Wang et al., 2025a; Bae and Kim, 2024; Xiong et al., 2025). These approaches mainly advance generation through better planning, memory, and execution.

However, the deeper bottleneck is the mismatch between training signals and reader experience. A chapter can be coherent yet unengaging, or a story can be structurally sound but lose emotional momentum. Tian et al. (2024) provide a systematic analysis of LLM-generated narratives and show that narrative arousal tends to decline in later story segments—a trajectory-level failure mode that current planning methods do not directly optimize. Meanwhile, LLM-as-a-judge methods are vulnerable to calibration bias and reward hacking in several settings (Gu et al., 2024; Zhao et al., 2025; Badshah and Sajjad, 2024).

To address these limitations, we propose **ChapterRL**, a chapter-aware reinforcement learning framework that explicitly reformulates multi-chapter novel generation as a **reward design problem**. Our core observation is that naturally occurring reader behavior on large-scale online fiction platforms contains rich chapter-level signals relevant to narrative quality. When these signals are extracted with principled debiasing, they become a scalable source of supervision. ChapterRL introduces two complementary reward signals. **Reading Depth Residual** (RDR) is a behavior-driven chapter-level reward constructed from reader continuation and completion behavior. It combines *continuation rate* and *completion rate*, then removes systematic genre and chapter-position confounds by stratified z-score normalization, yield-

ing a measure of *excess narrative attractiveness*. This signal answers the chapter-level question: *is this chapter compelling enough to sustain continued reading?*

Dynamic Rhythm Trajectory (DRT) specifically targets the later-story arousal decline identified by Tian et al. (2024). We first use DeepSeek-R1 as a teacher to annotate chapter-level arousal, distill these annotations into a lightweight Arousal Scorer, and then penalize chapters whose arousal falls substantially below the recent local mean. This threshold-gated design distinguishes normal narrative valleys from systematic rhythm collapse. DRT answers the story-level question: *does the story experience a systematic loss of emotional velocity during unfolding?*

The two signals are normalized within each prompt group and directly combined as the training reward. Our contributions are threefold:

1. We propose a **dual-level narrative reward framework** that is extensible beyond chapters to any ordered narrative segments, broadening its applicability to multi-stage creative writing.

2. We design a **behavior-based debiased reward construction method** (RDR) that converts large-scale reader continuation behavior into scalable chapter-level reward supervision.

3. We show empirically that ChapterRL improves narrative-specific benchmarks and strongly reduces late-story collapse, while RDR exhibits strong task-aligned preference accuracy on the Fiction and Scriptwriting domains of Chinese WritingPreferenceBench. Together with a segment-granularity ablation, these results suggest that ChapterRL is not limited to explicit chapter headers but can extend to multi-stage narrative writing.

2 Related Work

Reward Design for Creative Writing. The paradigm for improving creative text generation has been shifting from architectural innovations in planning and memory (Yang et al., 2023; Wang et al., 2025a; Yi et al., 2025; Xia et al., 2025; Wan et al., 2025) to the sophisticated design of reward functions for reinforcement learning (Yin et al., 2025; Gurung and Lapata, 2025). Recent works like LongWriter-Zero (Wu et al., 2025a) and Writing-RL (Lei et al., 2025) leverage composite rewards that combine metrics for general writing quality, length, and formatting. While effective, these frameworks target generic virtues like fluency and coherence (Liao et al., 2025), not the specific demands of multi-chapter fiction: sustained chapter-level engagement and coherent long-range narrative rhythm.

Limitations of Conventional Reward Sources. The dominant approaches for sourcing reward signals in creative writing face significant limitations. Human Preference-Based Reward Models, trained on pairwise comparisons (Jia et al., 2025; Li et al., 2025), are a standard approach. They effectively capture nuanced aspects of writing quality but suffer from high cost, limited scalability, and a tendency to converge towards "safe" preferences, stifling novelty. LLM-as-a-Judge and Generative Reward Models have emerged as scalable alternatives (Li et al., 2026; Lai et al., 2024). These methods use powerful LLMs to provide scores or detailed critiques, sometimes enhanced by structured rubrics (Gao et al., 2026) or multi-agent debate frameworks (Wang et al., 2025b). However, their reliability for long-form narrative evaluation is questionable. They are susceptible to systemic biases (e.g., position, verbosity) and often prioritize surface-level fluency over deep narrative substance (Gu et al., 2024; Ye et al., 2025; Xiao et al., 2024; Ahuja et al., 2025). Crucially, they struggle to track and penalize cumulative, long-horizon degradations such as the gradual collapse of narrative momentum, a key problem ChapterRL addresses.

User Behavior as a Debiased Reward Signal. A less-explored but promising alternative is to derive rewards from large-scale, natural user behavior, which reflects genuine reader experience (Toubia et al., 2021). ChapterRL follows this direction by converting large-scale reader behavior into debiased chapter-level reward supervision. Its Reading Depth Residual (RDR) component distills a clean signal of chapter-level narrative attractiveness from noisy engagement data (Zhou et al., 2024; Schmidt et al., 2023; Park and Kim, 2024). The Dynamic Rhythm Trajectory (DRT) component further penalizes systemic arousal decline in later story stages, a narrative-specific failure mode that generic quality rewards do not directly capture.

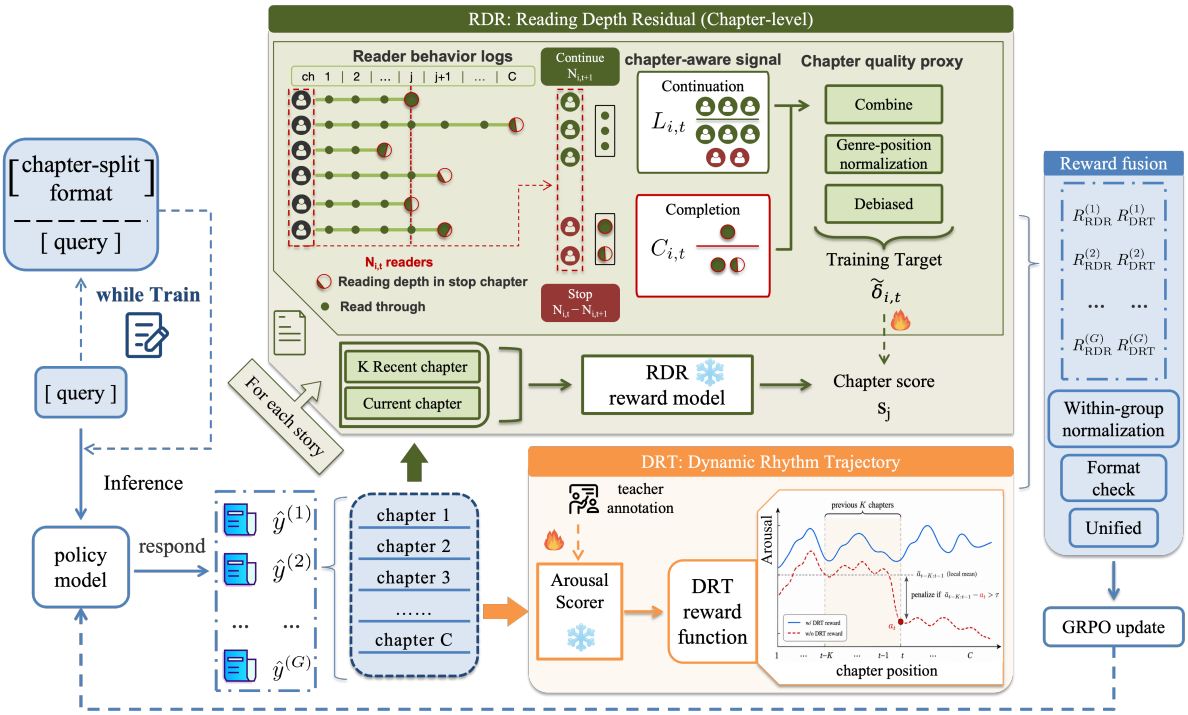

Figure 1: Overview of the **ChapterRL** framework. Given a multi-chapter novel prompt, the policy generates G structured responses, each segmented into an ordered chapter sequence $(r_1, \ldots, r_C)$. **Reading Depth Residual (RDR)** first constructs debiased chapter-level targets from reader continuation and completion behavior through genre–position stratified normalization. It then scores each target chapter using an M-chapter sliding window, predicting a scalar reward for the final chapter. **Dynamic Rhythm Trajectory (DRT)** suppresses systematic arousal decline through a threshold-gated local-window penalty, yielding R_{DRT}. The two signals are separately normalized within each prompt group, fused as the final reward, and used for GRPO training of the policy model, with a format check to enforce valid multi-chapter outputs.

3 Method

ChapterRL is a chapter-aware reinforcement learning framework for multi-chapter novel generation, where the reward unit is an ordered narrative segment and defaults to a natural chapter.

We decompose narrative quality into two complementary signals: Reading Depth Residual (RDR), derived from debiased continuation and completion behavior, and Dynamic Rhythm Trajectory (DRT), which penalizes later-story arousal collapse.

3.1 Problem Formulation

Let $\mathcal{D} = \{q_i\}$ denote a set of story-writing prompts. A policy model π_θ generates a structured response $\hat{y}$. We use a segmentation function $\mathcal{S}_\ell$ to convert the response into an ordered sequence of narrative units:

$$\mathcal{S}_\ell(\hat{y}) = [\hat{r}_1, \ldots, \hat{r}_C], \tag{1}$$

where ℓ denotes the segmentation granularity, $\hat{r}_j$ is the j-th generated narrative unit, and C is the number of units after segmentation. In the chapter-structured configuration, $\ell = \text{CHAPTER}$ and the units are natural chapters segmented by the instructed Chinese chapter headers, rendered as "Chapter 1", "Chapter 2", etc. in the English translation. More generally, ℓ can also correspond to fixed-length text blocks, such as 200-, 500-, or 1000-character segments. We train the policy using Group Relative Policy Optimization (GRPO) (Shao et al., 2024).

3.2 Reading Depth Residual

3.2.1 Behavioral data and quality proxies

We construct RDR labels from private reader-behavior logs collected from a large-scale online reading platform. The dataset contains over 20,000 web novels, 2 million chapters, and more than 10 million user reading interactions. From these logs, we construct over 700,000 RDR-labeled samples. For each story i, let C_i denote its number of chapters. For each reader whose terminal chapter is t, the data records the reading depth within that terminal chapter, denoted $d_{u,t} \in [0, 1]$. We use this terminal reading depth to construct a binary completion indicator for the terminal chapter: readers who finish the terminal

chapter are counted as completed, while readers who leave before finishing it are counted as not completed. From these logs, we derive $N_{i,t}$, the number of readers reaching chapter t in story i. We set $N_{i,C_i+1} = 0$ and define

$$D_{i,t} = N_{i,t} - N_{i,t+1}, \quad 1 \le t \le C_i,$$

where $D_{i,t}$ is the number of readers whose terminal chapter is t. Let $F_{i,t}$ denote the number of these terminal readers who finish chapter t according to terminal reading depth. We compute labels only for chapters with $N_{i,t} > 0$.

The **continuation rate** captures *inter-chapter decision behavior*:

$$L_{i,t} = \frac{N_{i,t+1}}{N_{i,t}}, \quad 1 \le t < C_i. \tag{2}$$

The **completion rate** captures *within-chapter depth behavior* among terminal readers and is defined when terminal readers are observed:

$$P_{i,t} = \frac{F_{i,t}}{D_{i,t}}, \quad D_{i,t} > 0. \tag{3}$$

When $D_{i,t} = 0$, completion is unobserved; in the combined proxy below, the chapter quality proxy falls back to continuation.

We combine the observable signals into a raw chapter-level quality proxy as

$$Q_{i,t} = \begin{cases} \alpha L_{i,t} + (1-\alpha) P_{i,t}, & t < C_i,\ D_{i,t} > 0, \\ L_{i,t}, & t < C_i,\ D_{i,t} = 0, \\ P_{i,t}, & t = C_i \end{cases}, \tag{4}$$

where $\alpha \in [0,1]$ controls the trade-off between cross-chapter continuation and within-chapter completion. A larger α emphasizes inter-chapter reading momentum, while a smaller α emphasizes local chapter completion. We set $\alpha = 0.7$, which emphasizes inter-chapter continuation while retaining completion-based satisfaction signals. Sensitivity results are provided in Appendix A.

3.2.2 Stratified debiasing normalization

Raw engagement signals contain two major confounds: **chapter-position effects** and **genre-popularity effects**. For example, first chapters naturally receive higher continuation rates than late chapters, and popular genres may receive stronger engagement regardless of chapter quality. We therefore apply genre–position stratified z-score normalization:

$$\tilde{\delta}_{i,t} = \mathrm{clip}\left(\frac{Q_{i,t} - \mu_{b(i,t)}}{\max\{\sigma_{b(i,t)}, \epsilon_\sigma\}}, -3, 3 \right), \tag{5}$$

where $b(i,t)$ denotes the selected normalization bucket: the genre–position bucket when it contains at least 30 samples, and the global position bucket otherwise. $\mu_{b(i,t)}$ and $\sigma_{b(i,t)}$ are the mean and standard deviation of $Q_{i,t}$ values in bucket $b(i,t)$, $\epsilon_\sigma > 0$ prevents division by zero, and $\mathrm{clip}(x, -3, 3)$ truncates x to $[-3, 3]$.

3.2.3 Reward model

We train RDR as a pointwise regression reward model to preserve the magnitude of the continuous residual label $\tilde{\delta}_{i,t}$ and reduce reward inference to one forward pass per candidate response.

M-chapter sliding-window scoring. For each story i with chapters $(\mathrm{Ch}_{i,1}, \ldots, \mathrm{Ch}_{i,C_i})$, we construct one reward-model instance for every target chapter t. Let M denote the sliding-window length. The input window contains the previous $M-1$ chapters and the target chapter:

$$\tilde{\mathrm{Ch}}_{i,k} = \begin{cases} [\mathrm{CHAPTER}]\ \langle \mathrm{empty} \rangle, & k < 1, \\ [\mathrm{CHAPTER}]\ \mathrm{Ch}_{i,k}, & k \ge 1 \end{cases}. \tag{6}$$

Here $[\mathrm{CHAPTER}]$ is a chapter separator and $\langle \mathrm{empty} \rangle$ is an empty padding placeholder. Early chapters are padded with empty slots to maintain a consistent window size.

The RDR reward model takes the M-chapter window as input and predicts a single scalar score for the final target chapter:

$$\hat{Q}_{i,t} = \mathrm{RDR}\left(X_{i,t}^{(M)} \right), \tag{7}$$

where $\hat{Q}_{i,t}$ is the predicted debiased attractiveness score of chapter t. The model is trained to regress the debiased reader-behavior label $\tilde{\delta}_{i,t}$:

$$\mathcal{L}_{\mathrm{RDR}} = \frac{1}{|\mathcal{T}|} \sum_{(i,t) \in \mathcal{T}} \mathrm{Huber}_{0.5} \left(\hat{Q}_{i,t} - \tilde{\delta}_{i,t} \right), \tag{8}$$

where $\mathcal{T}$ denotes the set of target chapters and $\mathrm{Huber}_{0.5}$ is the Huber loss with transition point 0.5. We use $M = 3$ in the final configuration, with sensitivity results reported in Appendix A.

At inference time, we apply the same sliding-window construction to the generated chapter sequence, and let $\hat{Q}_j = \mathrm{RDR}(\hat{X}_j^{(M)})$ denote the

Algorithm 1 DRT module pipeline

Require: Chapter sequence $\{r_t\}_{t=1}^{C}$, local window K, threshold τ
Ensure: Story-level DRT reward $R_{\mathrm{DRT}}(\hat{y})$
Use DeepSeek-R1 to identify the protagonist and extract protagonist-centered English emotion adjectives
Map the adjectives to NRC-VAD arousal values and average them into offline chapter-level labels $\tilde{a}_t$
Distill these labels into an Arousal Scorer on Chinese chapter text
Use the Arousal Scorer to predict chapter arousal scores a_t for generated story $\hat{y}$
for $t = 2, 3, \ldots, C$ **do**
    Compute the local mean $\bar{a}_{t-K:t-1}$ and gated penalty $p_t = \max(0, \bar{a}_{t-K:t-1} - a_t - \tau)$
end for
if $C < 2$ **then**
    Return 0
else
    Return $R_{\mathrm{DRT}}(\hat{y}) = -\frac{1}{C-1}\sum_{t=2}^{C} p_t$
end if

predicted score of the j-th generated target chapter, where $\hat{X}_j^{(M)}$ is its M-chapter generated window. For valid segmented outputs with $C \geq 1$, the story-level RDR reward is the average target-chapter score:

$$R_{\mathrm{RDR}}(\hat{y}) = \frac{1}{C}\sum_{j=1}^{C} \hat{Q}_j. \tag{9}$$

3.3 Dynamic Rhythm Trajectory

3.3.1 Arousal annotation and distillation

We first obtain offline chapter-level arousal labels using DeepSeek-R1 (DeepSeek-AI, 2025). For each chapter, the teacher follows a two-step annotation procedure: it first identifies the protagonist or central viewpoint character, and then extracts English emotion adjectives that trace the protagonist's emotional transitions. These descriptors are mapped to the NRC-VAD lexicon (Mohammad, 2018) and averaged over the arousal dimension to obtain a teacher label $\tilde{a}_t \in [0, 1]$. The exact Chinese prompt templates used for protagonist detection and protagonist-centered emotion extraction are provided in Appendix C. The NRC-VAD lexicon provides human-annotated valence, arousal, and dominance scores (each in $[0, 1]$) for over 20,000 English words, constructed via best–worst scaling.

To support online RL training, we distill these teacher-derived labels into a lightweight Arousal Scorer (AS). The distilled scorer achieves strong agreement with human annotations on a held-out test set of 363 books, with $\mathrm{MAE} = 0.084$, Pearson $r = 0.78$, and Spearman $\rho = 0.76$.

3.3.2 Reward formulation

Although we describe DRT using chapter terminology for the chapter-structured configuration, the formulation only requires the ordered sequence $\mathcal{S}_\ell(\hat{y})$. Thus, $a_t = \mathrm{AS}(\hat{r}_t) \in [0, 1]$ denotes the Arousal Scorer output for the t-th narrative unit, which can be either a natural chapter or a fixed-length segment in the granularity ablation.

For the t-th narrative unit with $t \geq 2$, given local-window length K and threshold τ, the gated DRT penalty is

$$p_t = \max\left(0, \bar{a}_{t-K:t-1} - a_t - \tau\right), \tag{10}$$

where the local mean over the previous K units is

$$\bar{a}_{t-K:t-1} = \frac{1}{\min(K, t-1)} \sum_{j=\max(1,t-K)}^{t-1} a_j. \tag{11}$$

The story-level DRT reward is the normalized negative mean penalty:

$$R_{\mathrm{DRT}}(\hat{y}) = \begin{cases} -\frac{1}{C-1}\sum_{t=2}^{C} p_t, & C \geq 2, \\ 0, & C = 1 \end{cases}. \tag{12}$$

The threshold τ is critical, as it penalizes only significant arousal drops, thus preserving natural narrative variations rather than forcing monotonic high intensity.

3.4 Unified Reward and Training

Because RDR and DRT operate on different scales, we standardize each component within the same prompt group. For a prompt group $\{\hat{y}^{(m)}\}_{m=1}^{G}$ and reward component $\kappa \in \{\mathrm{RDR}, \mathrm{DRT}\}$, let μ_κ and σ_κ denote the mean and standard deviation of $\{R_\kappa(\hat{y}^{(m)})\}_{m=1}^{G}$. We define

$$\bar{R}_\kappa^{(m)} = \frac{R_\kappa(\hat{y}^{(m)}) - \mu_\kappa}{\sigma_\kappa + \epsilon}. \tag{13}$$

where $\epsilon > 0$ is a small constant for numerical stability. The final reward for valid outputs is

$$\mathrm{Reward}(\hat{y}^{(m)}) = \bar{R}_{\mathrm{RDR}}^{(m)} + \bar{R}_{\mathrm{DRT}}^{(m)}. \tag{14}$$

This separate standardization makes the two components comparable without additional fusion weights.

Format penalty. In the main chapter-structured setting, we parse each generated response for the instructed chapter-header pattern, rendered as "Chapter 1", "Chapter 2", etc. in the English translation. If fewer than two chapters are detected—i.e., the model fails to follow the instructed multi-chapter format—we apply a fixed format penalty of -1.0 in place of the composite reward, strongly discouraging unstructured outputs. For fixed-length segment ablations, segmentation is performed after generation by character count, so the chapter-header format penalty is applied only to the main chapter-structured setting.

4 Experiments

We organize the empirical evaluation into three parts: datasets, training setup, and benchmarks and baselines.

4.1 Datasets

For RL training, we use a self-instruct paradigm with an LLM to generate 840 story synopses across 12 genres. Each synopsis is wrapped with one of two writing templates: 85% use the long prompt template and 15% use the short prompt template. Both templates require a multi-chapter format and a complete emotional arc, forming the final GRPO training set. (Wang et al., 2023b)

For reward construction, RDR is trained on chapter-level reader-behavior labels derived from continuation and completion statistics from a private fiction platform, while the Arousal Scorer is trained on DeepSeek-derived chapter-level arousal labels. The latter is used only to instantiate the DRT reward during RL.

4.2 Training Setup

RL training uses GRPO (Shao et al., 2024) with Qwen3-8B (Yang et al., 2025) as the policy model, $G = 8$, batch size 32, KL penalty coefficient $\beta = 0.01$, maximum generation length 32,000 tokens, and 3 training epochs (approximately 80 policy update steps). We also report a Qwen3-32B ChapterRL run under the same reward design for scaling comparison. The RDR reward model is trained as a pointwise scalar scorer on chapter-level reader-behavior residual labels. It uses the same M-chapter sliding-window construction described above, with empty-prefix padding for early chapters and a single supervision label for the final target chapter.

The Arousal Scorer uses Qwen3-8B with LoRA adapters ($r = 16$, $\alpha_{\text{LoRA}} = 32$, dropout 0.05, applied to attention query/value projections), trained for 5 epochs on DeepSeek-derived chapter-level arousal labels. For DRT, we use gating threshold $\tau = 0.05$ and local-window length $K = 3$. Both RDR and AS are frozen during RL. Metrics for trainable settings are averaged over three random seeds.

4.3 Benchmarks and Baselines

To evaluate multi-chapter novel generation, we use three automatic benchmarks for the final policy and one external preference benchmark for reward-model validation. The automatic benchmarks include Arena-Hard v2.0 Creative Writing (AH-CW), WebNovelBench, and WritingBench (Li et al., 2024; LM Arena Team, 2025; Lin et al., 2025; Wu et al., 2025b). All three automatic benchmarks use Claude-4-Sonnet as the evaluation model.

In addition, we evaluate the RDR reward model on the Chinese subset of WritingPreferenceBench (WPB) (Ying et al., 2025). Since RDR is designed to estimate chapter-level narrative attractiveness rather than general-purpose writing quality, we focus on the Fiction and Scriptwriting domains as task-aligned narrative slices. We use the published Chinese reward-model results from WPB as comparison baselines. This WPB evaluation validates the reward model itself, rather than the final ChapterRL policy.

Our baselines include strong proprietary models (GPT-4o (OpenAI, 2024) and Gemini-2.5-Pro (Comanici et al., 2025) under zero-shot prompting), standard backbone baselines (Qwen3-8B/32B zero-shot and Qwen3-8B + SFT), and two Qwen3-8B single-component ablations (GRPO + RDR only and GRPO + DRT only), together with the full Qwen3-32B ChapterRL setting (Yang et al., 2025). As generic RL references, we include RLMR's reported WritingBench scores (Liao et al., 2025).

5 Analysis Experiments

We analyze the final policy results (§5.1), later-story collapse (§5.2), RDR reward-model validation (§5.3), segmentation granularity and hyperparameter sensitivity (§5.4), and training dynamics (§5.5).

Model	Method	*Benchmark Quality*			*Collapse Metrics*	
		AH-CW ↑	WebNovel ↑	WritingBench ↑	LC Rate ↓	2H Drop ↓
Closed-source LLMs						
GPT-4o	Zero-shot	72.4	3.54	7.97	–	–
Gemini-2.5-Pro	Zero-shot	90.8	4.49	8.09	–	–
Other related work						
Qwen2.5-32B	GRPO + Writing RM	–	–	7.05	–	–
Qwen3-8B	RLMR	–	–	8.13	–	–
Open-source baselines						
Qwen3-8B	Zero-shot	18.1	3.37	7.05	51.3	0.053
Qwen3-8B	SFT	23.8	3.52	7.24	49.2	0.047
Qwen3-32B	Zero-shot	49.8	4.59	7.49	52.1	0.049
ChapterRL variants						
Qwen3-8B	GRPO + RDR only	47.6	4.61	7.61	38.8	0.032
Qwen3-8B	GRPO + DRT only	50.1	4.53	7.65	18.7	0.024
Qwen3-8B	**GRPO + RDR + DRT**	**52.8**	**4.73**	**8.01**	**15.0**	**0.011**
Qwen3-32B	GRPO + RDR + DRT	70.6	4.81	8.36	8.1	0.01
Real novels (reference)		–	–	–	1.3	0.008

Table 1: **Main results.** AH-CW is evaluated on the Creative Writing subset of Arena-Hard v2.0 (Li et al., 2024; LM Arena Team, 2025), using pairwise win rate (%) against responses generated by `gemini-2.0-flash` (50% indicates tie). WebNovelBench reports the overall narrative-quality score (Lin et al., 2025). WritingBench reports the overall writing-quality score on a 0–10 scale over the full benchmark (Wu et al., 2025b). All automatic benchmark evaluations for our reported baselines and ChapterRL variants use Claude-4-Sonnet as the evaluator. Results for the closed-source baselines, Qwen3 zero-shot baselines, SFT baseline, and ChapterRL variants are obtained under the same evaluation protocol. RLMR-related rows are external WritingBench reference results reported by Liao et al. (2025): the Qwen2.5-32B row corresponds to its Writing-RM-only GRPO baseline, while the Qwen3-8B row corresponds to its full mixed-reward RLMR result. They are included as related-work references because AH-CW, WebNovelBench, and collapse metrics are not reported in that work. LC Rate is the fraction of stories with significant late-stage collapse (%), and 2H Drop is the arousal drop from the first half to the second half. Results for our own trainable models are averaged over three random seeds unless otherwise specified.

5.1 Main Results

Table 1 presents the main results. Among Qwen3-8B variants, ChapterRL achieves the best overall performance, reaching 52.8 on AH-CW, 4.73 on WebNovelBench, 8.01 on WritingBench, 15.0% LC Rate, and 0.011 2H Drop.

Compared with Qwen3-8B zero-shot, ChapterRL improves by +34.7 on AH-CW, +1.36 on WebNovelBench, and +0.96 on WritingBench. Compared with Qwen3-8B + SFT, the gains are +29.0, +1.21, and +0.77. RDR-only gives the stronger WebNovelBench score, while DRT-only gives stronger collapse mitigation; combining them yields the best Qwen3-8B results across all reported metrics.

Qwen3-8B ChapterRL also outperforms Qwen3-32B zero-shot on all three benchmark-quality metrics. With the same reward design, Qwen3-32B ChapterRL further reaches 70.6 on AH-CW, 4.81 on WebNovelBench, and 8.36 on WritingBench.

WritingBench is a benchmark that includes a diverse range of creative text types. Since ChapterRL specializes in writing with chapter-based structures, its performance on WritingBench is less prominent. In contrast, WebNovelBench is a benchmark focused purely on novel writing, where ChapterRL demonstrates a more outstanding performance.

5.2 Later-Story Collapse Analysis

Table 1 and Figure 2 show that DRT directly improves late-stage rhythm stability. Qwen3-8B zero-shot has 51.3% LC Rate and 0.053 2H Drop. RDR-only reduces them to 38.8% and 0.032, while DRT-only reduces them further to 18.7% and 0.024. The full ChapterRL model obtains the best generated-story collapse metrics, with 15.0% LC Rate and 0.011 2H Drop.

5.3 RDR Reward Model Validation

Table 2 analyzes whether genre–position stratified normalization reduces chapter-position bias in RDR. We report the absolute Spearman correlation between chapter position and RDR score, together with the downstream WebNovelBench score.

We further evaluate RDR on the Chinese WPB

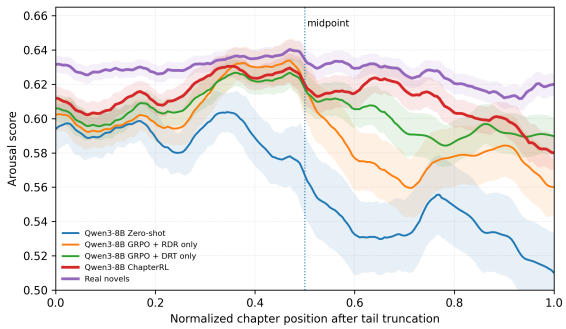

Figure 2: Arousal trajectories over normalized chapter position after tail truncation. ChapterRL maintains the most stable generated trajectory, while zero-shot shows the clearest late-stage decline. Shaded regions denote ± 2 standard errors.

Condition	$\lvert\rho_{\text{pos}}\rvert \downarrow$	**WebNovel** $\uparrow$
w/o norm.	0.35	4.38
w/ norm. (Ours)	0.06	**4.61**

Table 2: Position-bias analysis of the RDR reward. $\lvert\rho_{\text{pos}}\rvert$ is the absolute Spearman correlation between chapter position and RDR score; lower values indicate weaker position bias. The WebNovelBench score of the normalized setting is copied from the Qwen3-8B + GRPO + RDR-only result in Table 1.

Model	**Fiction**	**Script.**	**Narr. Avg.**
Scalar-output reward models			
Nvidia/AceMath-7B	54.9	54.6	54.8
RM-Mistral-7B	55.9	72.7	64.3
Skywork-Llama-8B	52.2	54.6	53.4
Skywork-Gemma-27B	53.3	45.5	49.4
RDR (Ours)	75.3	**81.8**	78.6
Generative reward models			
RM-R1-DS-Qwen-7B	56.5	36.4	46.5
RM-R1-DS-Qwen-14B	71.7	54.6	63.2
RM-R1-Qwen2.5-7B	**86.4**	72.7	**79.6**

Table 3: Task-aligned preference accuracy (%) of reward models on the Chinese subset of WritingPreferenceBench.

Fiction and Scriptwriting domains, which are most aligned with multi-stage narrative generation. Table 3 reports the results.

On the task-aligned WPB narrative domains, RDR achieves 75.3 on Fiction, 81.8 on Scriptwriting, and 78.6 on average. It outperforms the listed scalar-output reward models and remains competitive with the strongest generative reward model on narrative average.

5.4 Segment Granularity Ablation

We test whether ChapterRL depends on natural chapter boundaries by replacing chapter segmentation with fixed-length 200-, 500-, and 1000-character blocks, while keeping the policy model, reward models, training prompts, and GRPO configuration unchanged. Table 4 shows that natural chapters perform best, with 4.73 on WebNovelBench and 15.0% LC Rate. Among fixed-length settings, 500-character segmentation is strongest, reaching 4.61 and 18.6%.

Reward unit	**WebNovel** $\uparrow$	**LC Rate** $\downarrow$
200-char	4.32	37.2
500-char	4.61	18.6
1000-char	4.46	24.1
Natural chapter	**4.73**	**15.0**

Table 4: Segment-granularity ablation for ChapterRL.

5.5 Ablation and Sensitivity Analysis

Beyond segmentation granularity, ChapterRL has four design-relevant hyperparameters: α (RDR continuation–completion mixture weight), τ (DRT gating threshold), M (RDR sliding-window length), and K (DRT local window). We report the sensitivity results for these hyperparameters in Appendix A.

5.6 Training Dynamics and Reward Hacking

We monitor average generation length during RL training to check for length-based reward hacking. ChapterRL does not show significant length inflation, suggesting that the combined RDR and DRT rewards do not encourage simple length expansion.

6 Conclusion

We presented ChapterRL, a chapter-aware reinforcement learning framework that reframes multi-chapter novel generation as a reward design problem. By combining Reading Depth Residual (RDR), which extracts debiased supervision from reader continuation behavior, with Dynamic Rhythm Trajectory (DRT), which penalizes later-story arousal collapse through threshold-gated trajectory modeling, ChapterRL aligns training signals with reader experience at both local and global temporal scales. Although natural chapters are the default unit in our main setting, the segment-granularity analysis suggests that the framework can also operate over fixed-length narrative segments. Our WPB analysis further validates the RDR reward model as a task-aligned scalar preference scorer for narrative domains, including Scriptwriting, while the final ChapterRL policy is assessed through generation-quality and rhythm-collapse evaluations.

Limitations

Our RDR data is proprietary, which limits reproducibility. To mitigate this, we propose and validate an LLM-based method to simulate reader behavior as a public alternative (AppendixE). The RDR sliding-window length M presents a trade-off between broader narrative context and computational cost. The Arousal Scorer is trained and validated on Chinese literary text, so transferring it to other languages or domains would likely require re-distillation. Although our segment-granularity ablation suggests that ChapterRL can operate on fixed-length text segments, natural chapters remain the most effective reward unit in our experiments. Applying the framework to other multi-stage creative writing tasks, such as screenplays or interactive role-playing, may benefit from task-specific segmenters that better capture scenes, dialogue turns, or dramatic beats.

References

K. Ahuja, M. Sclar, and Y. Tsvetkov. 2025. Finding flawed fictions: Evaluating complex reasoning in language models via plot hole detection. *arXiv preprint*, arXiv:2504.11900.

S. Badshah and H. Sajjad. 2024. Reference-guided verdict: Llms-as-judges in automatic evaluation of free-form text. *Preprint*, arXiv:2408.01235.

Minwook Bae and Hyounghun Kim. 2024. Collective critics for creative story generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA. Association for Computational Linguistics.

John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. 2022. Talebrush: Sketching stories with generative pretrained language models. In *CHI Conference on Human Factors in Computing Systems*, CHI '22, pages 1–19.

Gheorghe Comanici and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *Preprint*, arXiv:2501.12948.

Fan Gao, Dongyuan Li, Ding Xia, Fei Mi, Yasheng Wang, Lifeng Shang, and Baojun Wang. 2026. EssayBench: Evaluating large language models in multi-genre chinese essay writing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 37396–37406.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2024. A survey on LLM-as-a-Judge. *Preprint*, arXiv:2411.15594.

Alexander Gurung and Mirella Lapata. 2025. Learning to reason for long-form story generation. *Preprint*, arXiv:2503.22828.

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. *Preprint*, arXiv:2506.00103.

Yuxiang Lai, Shiliang Wang, Siyuan Liu, Xuanjing Huang, and Zhongyu Wei. 2024. ALaRM: Align language models via hierarchical rewards modeling. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7817–7831. Association for Computational Linguistics.

Xuanyu Lei, Chenliang Li, Yuning Wu, Kaiming Liu, Weizhou Shen, Peng Li, Ming Yan, Ji Zhang, Fei Huang, and Liu Yang. 2025. Writing-RL: Advancing long-form writing via adaptive curriculum reinforcement learning. *Preprint*, arXiv:2506.05760.

J. Li, Y. Chen, Z. Liu, M. Tan, L. Zhang, Y. Li, and M. Yang. 2025. Storyteller: An enhanced plot-planning framework for coherent and cohesive story generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20818–20846. Association for Computational Linguistics.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *Preprint*, arXiv:2406.11939.

Zhaoyan Li, Hang Lei, Yujia Wang, Lanbo Liu, Hao Liu, and Liang Yu. 2026. Rewarding creativity: A human-aligned generative reward model for reinforcement learning in storytelling. *Preprint*, arXiv:2601.07149.

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. *Preprint*, arXiv:2508.18642.

Leon Lin, Jun Zheng, and Haidong Wang. 2025. WebNovelBench: Placing LLM novelists on the web novel distribution. *Preprint*, arXiv:2505.14818.

LM Arena Team. 2025. Arena-hard-auto: An automatic LLM benchmark. `https://github.com/lmarena/arena-hard-auto`. Official repository; Arena-Hard v2.0 includes 250 creative writing queries.

Saif M. Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184, Melbourne, Australia. Association for Computational Linguistics.

OpenAI. 2024. GPT-4o system card. *Preprint*, arXiv:2410.21276.

C. Park and J. Kim. 2024. Exploring affective representations in emotional narratives: An exploratory study comparing ChatGPT and human responses. *Cyberpsychology, Behavior, and Social Networking*, 27(10):736–741.

M. L. C. Schmidt, J. R. Winkler, M. Appel, and T. Richter. 2023. Tracking emotional shifts during story reception: The relationship between narrative structure and affective responses. *Scientific Study of Literature*, 12(1):1–26.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Yufei Tian, Tenghao Huang, Miri Liu, Derek Jiang, Alexander Spangher, Muhao Chen, Jonathan May, and Nanyun Peng. 2024. Are large language models capable of generating human-level narratives? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17659–17681, Miami, Florida, USA. Association for Computational Linguistics.

Olivier Toubia, Jonah Berger, and Jehoshua Eliashberg. 2021. How quantifying the shape of stories predicts their success. *Proceedings of the National Academy of Sciences*, 118(26):e2011695118.

K. Wan, H. Mu, R. Hao, H. Luo, T. Gu, and X. Chen. 2025. A cognitive writing perspective for constrained long-form text generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 9832–9844. Association for Computational Linguistics.

Qianyue Wang, Jinwu Hu, Zhengping Li, Yufeng Wang, Daiyuan Li, Yu Hu, and Mingkui Tan. 2025a. Generating long-form story using dynamic hierarchical outlining with memory-enhancement. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics.

Xinda Wang, Zhengxu Hou, Yangshijie Zhang, Bingren Yan, Zhibo Yang, Xingsheng Zhang, Luxi Xing, Qiang Zhou, and Chen Zhang. 2025b. EvolvR: Self-evolving pairwise reasoning for story evaluation to enhance generation. *Preprint*, arXiv:2508.06046.

Yichen Wang, Kevin Yang, Xiaoming Liu, and Dan Klein. 2023a. Improving pacing in long-form story planning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10788–10845, Singapore. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Zhihua Wen, Zhiliang Tian, Wei Wu, Yuxin Yang, Yanqi Shi, Zhen Huang, and Dongsheng Li. 2023. GROVE: A retrieval-augmented complex story generation framework with a forest of evidence. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3980–3998, Singapore. Association for Computational Linguistics.

Yuhao Wu, Yushi Bai, Zhiqiang Hu, Roy Ka-Wei Lee, and Juanzi Li. 2025a. LongWriter-Zero: Mastering ultra-long text generation via reinforcement learning. *Preprint*, arXiv:2506.18841.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Qin, and Fei Huang. 2025b. WritingBench: A comprehensive benchmark for generative writing. *Preprint*, arXiv:2503.05244.

Haotian Xia, Hao Peng, Yunjia Qi, Xiaozhi Wang, Bin Xu, Lei Hou, and Juanzi Li. 2025. StoryWriter: A multi-agent framework for long story generation. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 6559–6563. Association for Computing Machinery.

J. Xiao, Z. Li, X. Xie, E. Getzen, C. Fang, Q. Long, and W. J. Su. 2024. On the algorithmic bias of aligning large language models with RLHF: Preference collapse and matching regularization. *arXiv preprint*, arXiv:2405.16455.

Ruibin Xiong, Yimeng Chen, Dmitrii Khizbullin, Mingchen Zhuge, and Juergen Schmidhuber. 2025. Beyond outlining: Heterogeneous recursive planning for adaptive long-form writing with language models. *Preprint*, arXiv:2503.08275.

An Yang, Anfeng Li, Baosong Yang, and 1 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Kevin Yang, Dan Klein, Nanyun Peng, and Yuandong Tian. 2023. DOC: Improving long story coherence with detailed outline control. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3378–3465, Toronto, Canada. Association for Computational Linguistics.

J. Ye, Y. Wang, Y. Huang, D. Chen, Q. Zhang, N. Moniz, and X. Zhang. 2025. Justice or prejudice? quantifying biases in LLM-as-a-judge. In *International Conference on Learning Representations*, pages 102351–102390.

Q. Yi, Y. He, J. Wang, X. Song, S. Qian, X. Yuan, and J. Ni. 2025. Score: Story coherence and retrieval enhancement for AI narratives. *arXiv preprint*, arXiv:2503.23512.

Y. Yin, S. Yang, Y. Xie, Z. Yang, Y. Sun, H. Awadalla, and M. Zhou. 2025. Segmenting text and learning their rewards for improved rlhf in language model. *arXiv preprint*, arXiv:2501.02790.

Shuangshuang Ying, Yunwen Li, Xingwei Qu, Xin Li, Sheng Jin, Minghao Liu, Zhoufutu Wen, Xeron Du, Tianyu Zheng, Yichi Zhang, Letian Ni, Yuyang Cheng, Qiguang Chen, Jingzhe Ding, Shengda Long, Wangchunshu Zhou, Jiazhan Feng, Wanjun Zhong, Libo Qin, and 4 others. 2025. Beyond correctness: Evaluating subjective writing preferences across cultures. *Preprint*, arXiv:2510.14616.

Yulai Zhao, Haolin Liu, Dian Yu, Sunyuan Kung, Meijia Chen, Haitao Mi, and Dong Yu. 2025. One token to fool LLM-as-a-Judge. *Preprint*, arXiv:2507.08794.

H. Zhou, S. Muppalla, Z. S. Yildiz, M. Kim, A. Sahai, and N. Peng. 2024. Measuring psychological depth in language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17162–17196. Association for Computational Linguistics.

A Ablation and Sensitivity Details

This section provides additional ablation and sensitivity results for ChapterRL. We summarize the sensitivity of four main hyperparameters in Table 5: the RDR continuation–completion mixture weight α, the RDR sliding-window length M, the DRT local window K, and the DRT gating threshold τ.

Module	Model config	Hyperparameter	Setting	Metric	Value
RDR	*label mixture*				
RDR	Qwen3-8B + GRPO + RDR only	α	0.3	WebNovel ↑	4.38
RDR	Qwen3-8B + GRPO + RDR only	α	0.5	WebNovel ↑	4.43
RDR	Qwen3-8B + GRPO + RDR only	α	**0.7**	WebNovel ↑	**4.61**
RDR	Qwen3-8B + GRPO + RDR only	α	0.9	WebNovel ↑	4.17
RDR	*sliding-window length*				
RDR	Qwen3-8B + GRPO + RDR only	M	1	WebNovel ↑	4.25
RDR	Qwen3-8B + GRPO + RDR only	M	2	WebNovel ↑	4.59
RDR	Qwen3-8B + GRPO + RDR only	M	**3**	WebNovel ↑	**4.61**
RDR	Qwen3-8B + GRPO + RDR only	M	4	WebNovel ↑	4.27
DRT	*local-window length*				
DRT	Qwen3-8B + GRPO + DRT only	K	2	LC / 2H ↓	28.5% / 0.031
DRT	Qwen3-8B + GRPO + DRT only	K	**3**	LC / 2H ↓	**18.7% / 0.024**
DRT	Qwen3-8B + GRPO + DRT only	K	4	LC / 2H ↓	20.4% / 0.026
DRT	Qwen3-8B + GRPO + DRT only	K	5	LC / 2H ↓	19.3% / 0.026
DRT	*gating threshold*				
DRT	Qwen3-8B + GRPO + DRT only	τ_{gate}	0.00	LC / 2H ↓	17.1% / 0.025
DRT	Qwen3-8B + GRPO + DRT only	τ_{gate}	0.03	LC / 2H ↓	19.3% / 0.026
DRT	Qwen3-8B + GRPO + DRT only	τ_{gate}	**0.05**	LC / 2H ↓	**18.7% / 0.024**
DRT	Qwen3-8B + GRPO + DRT only	τ_{gate}	0.08	LC / 2H ↓	27.5% / 0.030

Table 5: Component-level sensitivity summary for key hyperparameters. RDR-related ablations are conducted with Qwen3-8B + GRPO + RDR only and evaluated by WebNovelBench. DRT-related ablations are conducted with Qwen3-8B + GRPO + DRT only and evaluated by LC Rate and 2H Drop, abbreviated as LC / 2H. The bold rows correspond to the single-component settings reported in Table 1; the full ChapterRL result is reported separately in Table 1.

B Prompt Construction Details

This appendix describes the GRPO prompt construction procedure. All operational prompts are written in Chinese, and the original prompt strings are provided as rendered figures.

Overview. Prompt construction consists of two stages. First, we generate diverse story synopses with a self-instruct procedure. Second, each accepted synopsis is wrapped with either a long or short writing template to form the final prompt consumed by the GRPO policy. The final prompt set uses an 85% long-template and 15% short-template mixture. The genre label and prompt type are retained as metadata for data balancing and analysis, but they are not inserted as additional messages.

No reward-related information is included in the prompt. In particular, reader-behavior statistics, RDR labels, DRT arousal labels, reward-model outputs, and evaluation annotations are never exposed to the policy model. The prompt only specifies the desired writing format and the story synopsis.

Prompt construction dataflow. Table 6 summarizes how genre specifications are converted into GRPO training instances. The exact Chinese prompt strings are provided in Figures 3, 4, and 5.

Genre-specific instructions. We use 12 predefined Chinese web-fiction genres. Each genre is associated with a short instruction that constrains the synopsis-generation model toward genre-appropriate settings, conflicts, and character relations. The exact genre keys and instruction strings used in the implementation are shown in Figure 3.

Stage	Input	Operation	Output	Policy-visible?
Genre specification	Genre names and genre instructions	Define genre-specific constraints for synopsis generation.	Genre-to-instruction mapping.	No
Synopsis generation	Genre, genre instruction, target count, and existing synopses	Query DeepSeek-R1 with a system message and a user message. Up to 10 existing same-genre synopses are injected as diversity context.	Candidate story synopses.	No
Filtering and deduplication	Candidate synopses	Parse numbered outputs and remove malformed items, exact duplicates, and near duplicates.	Accepted synopsis pool.	No
Prompt assembly	Accepted synopsis and prompt type	Select the corresponding format block and concatenate it with the synopsis.	Final user message.	Yes
Instance packaging	Final user message, prompt type, and genre	Store the final prompt under `messages`; retain `type` and `category` as metadata only.	GRPO training instance.	Partially

Table 6: GRPO prompt construction pipeline. Only `messages[0].content` is consumed by the policy model; `type` and `category` are metadata.

Code key	Instruction string
现代言情	都市背景、现代职场或生活场景，男女主角因某种偶然相遇，经历误会、挫折、成长最终走到一起
悬疑推理	充满谜题和反转的故事，有悬案、隐藏的秘密、出人意料的真相，推理逻辑严密
都市情感	现代都市生活，聚焦情感羁绊：友情、亲情、暗恋、失恋、职场人际等真实生活困境
幻想言情	奇幻或玄幻世界观，有特殊力量或设定，男女主角跨越种族/阵营/命运的感情
历史军事	古代或近代历史背景，战争、权谋、家国情怀，人物在乱世中的成长与选择
武侠仙侠	江湖恩怨、门派纷争、修炼体系，侠义精神与感情线交织，有精彩的打斗场面
科幻末世	末世、未来、星际或反乌托邦设定，人类面临极端困境，展现生存意志与人性光辉
古代言情	古代背景、宫廷或民间，男女主角命运交错，有权谋博弈或身份差异带来的波折
都市悬疑	现代城市背景，结合悬疑与都市生活，普通人卷入事件，展现城市暗面与人性复杂
职场商战	职场竞争、商业博弈，主角从底层逆袭或在权力斗争中坚守初心，展现现实残酷
家庭伦理	家庭关系、代际冲突、婚姻矛盾，深度刻画人物内心，展现中国家庭的温情与羁绊
青春校园	校园生活、青春懵懂的情感、成长的烦恼，展现纯真岁月与青春的遗憾和美好

Figure 3: Exact Chinese genre keys and instruction strings used in `CATEGORY_INSTRUCTIONS`.

Stage 1: synopsis generation. For each genre, we prompt DeepSeek-R1 to generate new synopses conditioned on the genre instruction and a small set of existing same-genre examples. This context encourages plot-level diversity and reduces repetition.

The synopsis-generation prompt contains a system message and a user message. The system message specifies the model role, while the user message specifies the target genre, genre instruction, existing examples, output format, and the number of requested synopses. The exact Chinese strings used in this stage are shown in Figure 4.

Code field	Exact prompt template
system_msg	你是一个专业的中文网络小说策划，擅长为各类型小说设计新颖、具体、有吸引力的故事梗概。
user_msg	请为【{category}】类型小说生成 {count} 个不同的故事梗概。 类型要求：{instruction} 已有梗概（避免重复，请确保新梗概在情节设定上有明显区别）：{existing_samples} 要求： 1. 每个梗概 1-2句话，描述核心人物关系和矛盾冲突 2. 情节设定具体、新颖，避免套路化 3. 有明确的故事核心矛盾（非简单的相遇相爱） 4. 每行一个梗概，用数字编号，格式：1. [梗概内容] 5. 只输出梗概列表，不要其他说明文字 请生成 {count} 个：

Figure 4: Exact Chinese strings used by the synopsis-generation prompt constructor.

Generated outputs are parsed line by line according to the required numbered-list format. We remove malformed items, exact duplicates, and embedding-based near duplicates before adding accepted synopses to the pool. We target 70 accepted synopses per genre, resulting in approximately 840 prompts across 12 genres.

Stage 2: final GRPO prompt assembly. Each accepted synopsis is converted into a GRPO training prompt by applying one of two writing templates. The final prompt set contains 85% `long` prompts and 15% `short` prompts. The selected template is concatenated with the synopsis into a single user message. The exact Chinese strings for both templates and the final prompt format are shown in Figure 5.

Code field	Exact prompt template
FORMAT_BLOCK	【格式要求】 必须分章节展示，使用阿拉伯数字：第1章、第2章、第3章... 每章标题独占一行，例如： 第1章 [章节标题] [章节内容] 第2章 [章节标题] [章节内容]
SHORT_FORMAT_BLOCK	【格式要求】 篇幅简短，仅需生成一个独立情节片段或单章内容。 总字数约800-1500字。
content	请根据以下要求创作一篇故事：{format_req} 【故事梗概】{synopsis}

Figure 5: Exact Chinese strings used by the final GRPO prompt assembly function.

Final training instance schema. After assembly, each GRPO training instance has the following structure:

{

```
  "messages": [
    {
      "role": "user",
      "content": "<final writing prompt>"
    }
  ],
  "type": "long_or_short",
  "category": "<genre>"
}
```

Only `messages[0].content` is used as input to the policy model. The fields `type` and `category` are metadata and are not exposed as additional messages during GRPO rollouts.

C DRT Arousal Annotation Prompt

This appendix provides the prompt details for the offline arousal annotation step used by Dynamic Rhythm Trajectory (DRT). The purpose of this step is to obtain chapter-level arousal supervision from DeepSeek-R1 before training the Arousal Scorer. During GRPO training, DeepSeek-R1 is not queried; the frozen Arousal Scorer is used instead.

Two-step annotation procedure. For each chapter, we use a two-step DeepSeek-R1 annotation pipeline. First, the teacher identifies the protagonist or central viewpoint character of the chapter. Second, conditioned on the detected protagonist, the teacher extracts protagonist-centered English emotion adjectives. These adjectives are intended to capture the protagonist's emotional transitions within the chapter. We use English adjectives because the subsequent arousal lookup is performed with the NRC-VAD lexicon.

The operational prompts are written in Chinese; Figure 6 provides the protagonist-detection and protagonist-centered emotion-extraction templates.

Stage	Exact prompt template
Protagonist detection	以下是一段中文小说章节。本章的主角是谁？（即视角最集中、经历与情感最突出的人物） 输出应只包含人物姓名，不要输出任何其他内容。
Protagonist-centered emotion extraction	以下是一段中文小说章节，主角是「`{protagonist}`」。 请追踪主角「`{protagonist}`」的情感变化，每当情感发生明显转变时记录一个新阶段。 你的任务：为每个阶段输出**一个英文形容词**要求： 1. 只描述主角「`{protagonist}`」的情感，忽略其他人物。 2. 选词需同时体现情感**类别**和**强度**—— 例如同是「悲」：grieving（极度）/ sorrowful（中度）/ melancholy（轻度）； 例如同是「怒」：furious（极度）/ angry（中度）/ irritated（轻度）。 3. 优先选常见英文情感形容词。 4. 只输出一个 JSON 数组，不含任何解释例如：`["calm", "curious", "anxious", "angry", "sad"]`

Figure 6: Exact Chinese prompt templates used for the DeepSeek-based DRT arousal annotation pipeline. The first row shows protagonist detection, and the second row shows protagonist-centered emotion adjective extraction.

Prompt inputs and outputs. In the protagonist-detection stage, the chapter text is appended after the prompt template, and the teacher is instructed to output only the protagonist name. In the emotion-extraction stage, the detected protagonist name is inserted into the prompt, and the chapter text is appended after the template. The teacher is required to output a JSON array of English emotion adjectives, such as `["calm", "curious", "anxious", "angry", "sad"]`, without any additional explanation.

From emotion adjectives to arousal labels. The extracted English emotion adjectives are lowercased and mapped to NRC-VAD arousal values. For a nonempty multiset A_t of matched arousal values, we average them into the chapter-level teacher label:

$$\tilde{a}_t = \frac{1}{|A_t|} \sum_{v \in A_t} v, \tag{15}$$

where A_t is the multiset of NRC-VAD arousal values matched from the extracted emotion adjectives of chapter t. The resulting scalar label $\tilde{a}_t \in [0, 1]$ is used as the supervision target for training the Arousal Scorer.

Fallback behavior. If protagonist detection fails, a generic protagonist placeholder is used. If emotion extraction fails, the teacher output cannot be parsed as a valid JSON array, or no extracted adjective can be matched to NRC-VAD, we assign the neutral teacher label $\tilde{a}_t = 0.5$. This conservative fallback prevents failed annotations from introducing artificially high or low arousal labels.

D Hyperparameter Details

Table 7 summarizes key hyperparameters used in RDR reward model training, Arousal Scorer training, and GRPO policy training.

Component	Parameter	Value
RDR RM	Tuning method	Full fine-tuning + ZeRO-3
RDR RM	Learning rate	5.0×10^{-6}
RDR RM	Effective batch size	256
RDR RM	Max sequence length	8,192
RDR RM	Epochs	5
RDR RM	Huber δ	0.5
Arousal Scorer	Tuning method	LoRA (q_proj, v_proj)
Arousal Scorer	LoRA rank r	16
Arousal Scorer	LoRA α_{LoRA}	32
Arousal Scorer	LoRA dropout	0.05
Arousal Scorer	Learning rate	2.0×10^{-5}
Arousal Scorer	Effective batch size	64
Arousal Scorer	Max sequence length	2,048
Arousal Scorer	Epochs	5
GRPO (policy)	G (rollouts per prompt)	8
GRPO (policy)	Batch size (prompts per step)	32
GRPO (policy)	KL coefficient β	0.01
GRPO (policy)	Max generation length	32,000 tokens
GRPO (policy)	Learning rate	1.0×10^{-6}
GRPO (policy)	Training epochs	3 (approx. 80 steps)
DRT	Local window K	3
DRT	Threshold τ	0.05
RDR	Sliding-window length M	3
RDR	Mixture weight α	0.7

Table 7: Hyperparameter summary. RDR uses full-parameter fine-tuning with DeepSpeed ZeRO-3 due to large training data scale, while the Arousal Scorer uses LoRA adapters due to comparatively limited teacher-annotated data.

E Prompt for Simulating Reader Behavior

Data Source	AH-CW	WebNovel	WritingBench	Fiction	Script.
constructed data	50.5	4.47	7.79	70.2	78.9
real data	**52.8**	**4.73**	**8.01**	**75.3**	**81.8**

Table 8: The performance of ChapterRL using our constructed data versus real data.

Because the RDR training data comes from a proprietary reading platform, we provide an LLM-based alternative for constructing per-chapter continuation and completion rates. This simulated reader-behavior data can be used to train the RDR module when real platform logs are unavailable.

The prompts are written in Chinese, and the simulation template is shown in Figure 7.

Table 8 compares the performance of ChapterRL (based on Qwen3-8B) using our constructed data versus real data, and it shows a similar comparison for the RDR module on the Chinese WPB Fiction and Scriptwriting domains. The results show that simulated reader-behavior data preserves most of the gains obtained from real platform logs.

Code field / stage	Original Chinese template	Code field / stage	Original Chinese template

SYSTEM_PROMPT	你是一个权威的小说章节质量评估专家。你的评估标准不受小说题材类型影响——无论悬疑、言情、科幻还是其他类型，你只关注其文学叙事质量，包括：情节推进是否合理、人物是否有吸引力、节奏是否舒适、是否有让读者想继续阅读的驱动力。
	
	你的任务：通读一本小说的所有章节，进行质量排序，然后为每章输出两个核心指标。
		QUALITY_TASK_TEMPLATE / output_format	严格输出一个 JSON 对象，不要包含 markdown 代码块标记：
			{
			 "book_id": {book_id},
			 "chapter_count": {chapter_count},
			 "overall_assessment": "对全书整体质量的简短评价（1-2句）",
			 "chapter_scores": [{
			 "chapter_idx": 1,
			 "retention_rate": 0.88,
			 "completion_depth": 1.0,
			 "quality_rank": 3,
			 "notes": "可选：简述该章质量判断依据"
			 }]
			 }
			
			注意：chapter_idx 从 1 到 {chapter_count}; quality_rank 中 1 表示质量最高，{chapter_count} 表示质量最低; retention_rate 和 completion_depth 都保留 2 位小数；必须覆盖全部章节。

CHAPTER_INPUT_FORMAT	## 第{i + 1}章: {chapter_title} {chapter_text} 若单章过长，仅保留章节前部文本以控制输入长度。	QUALITY_TASK_TEMPLATE / chapter_payload	以下是本书的全部章节内容： {chapters_text}
QUALITY_TASK_TEMPLATE / step_1	请按以下步骤完成质量评估： 步骤 1：整体排序 通读以下 {chapter_count} 个章节的文本，将它们按“读者阅读驱动力的强弱”从高到低排序。 驱动力的定义：读者读完本章后想继续读下一章的欲望强度，综合考虑情节进展、悬念设置、情感张力、信息增量等。	MULTI_SEGMENT_PROMPT / first_segment	小说信息：书名 {book_name}；总章节数 {chapter_count}；总字数 {total_words}。 由于书籍较长，会分段发送。这是第 1 段，共 {total_segments} 段，包含第 1-{first_segment_end} 章。请阅读以下章节内容，记住每章的质量水平，暂时不要评价： {first_segment_text} 请简要回复“已阅读第 1-{first_segment_end} 章，等待后续章节”即可。
QUALITY_TASK_TEMPLATE / step_2	步骤 2：质量评分 对你完成的排序，请为每章输出 retention_rate（续读率）和 completion_depth（完读深度）。 retention_rate（续读率）：若 100 位读者读到本章末尾，约有多少人会继续读下一章。 completion_depth（完读深度）：在被本章内容劝退、停在本章的读者中，他们平均读完了本章的多少比例。	MULTI_SEGMENT_PROMPT / middle_segment	这是第 {segment_idx} 段，共 {total_segments} 段，包含第 {segment_start}-{segment_end} 章： {segment_text} 请继续阅读，简要回复“已阅读第 {segment_start}-{segment_end} 章，等待后续章节”即可。
QUALITY_TASK_TEMPLATE / score_anchors	retention_rate 评分锚定： - 0.95 = 章节有强悬念或情感高潮驱动，几乎无人愿意在此停下 - 0.85 = 质量不错的章节，大部分读者会继续 - 0.75 = 质量中等的章节，约 1/4 读者可能流失 - 0.65 = 章节有明显质量下降，约 1/3 读者会流失 - 0.50 = 章节拖沓、冗余或无实质推进，约一半读者可能放弃 completion_depth 评分锚定： - 1.0 = 读完了整章，但读完决定不继续下一章 - 0.7 = 读到约 70% 处放弃 - 0.4 = 读到不到一半放弃 - 0.15 = 几乎立刻放弃，只读了开头	MULTI_SEGMENT_PROMPT / final_segment	这是最后一段，第 {total_segments} 段，共 {total_segments} 段，包含第 {last_start}-{last_end} 章： {last_segment_text} 以下为全书所有 {chapter_count} 章的标题摘要，供你回顾全书结构： {all_chapter_titles} 现在你已经读完全部 {chapter_count} 章，请按与单轮提示相同的流程完成质量评估：整体排序、章节评分、质量约束，并严格输出 JSON。
QUALITY_TASK_TEMPLATE / constraints	步骤 3：质量约束 - 排序相邻的章节，retention_rate 差至少 ≥ 0.02，除非确实几乎无差异。 - 每本书的 retention_rate 最高分不得超过 0.95，最低分不得低于 0.40。 - 对于质量确实接近的章节，允许相同或接近的分数，但需在 notes 中说明原因。	FINAL_OUTPUT	每章得到结构化读者行为代理数据：chapter_idx、retention_rate、completion_depth、quality_rank、notes。

Figure 7: Prompt for Simulating Reader Behavior.