GoLongRL: Capability-Oriented Long-Context Reinforcement Learning with Heterogeneous Multitask Alignment

Anonymous authors
Paper under double-blind review

Abstract

Reinforcement learning with verifiable rewards (RLVR) has shown promise for improving how large language models use long contexts. Existing long-context RL methods, however, commonly construct data by increasing retrieval-path complexity, leaving capabilities such as aggregation, structured extraction, ranking, ordering, and summarization with limited direct supervision. We present GoLongRL, a capability-oriented framework for long-context RLVR. Guided by the capability dimensions of LongBench Pro, we construct a 23K-sample dataset spanning nine task types, each paired with a reward that matches its output semantics. The dataset combines curated long-context annotations with question–answer pairs generated from real documents and applies multi-stage quality verification. With GRPO, training on the constructed dataset raises the average over six long-context benchmarks from 53.0 to 62.2 at 4B and from 60.1 to 69.8 at 30B. At both scales, these results exceed those obtained by GRPO trained on data constructed by QwenLong-L1.5, while the trained models also show robust performance on general reasoning, agentic memory, dialogue memory, and overlong-context evaluations. At the same time, optimizing heterogeneous task-specific rewards creates an additional challenge because their different scales induce inconsistent advantage scales across tasks. GoLongRL therefore introduces TMN-Reweight, which combines task-level mean normalization with difficulty-adaptive reweighting to align cross-task scales and account for prompt difficulty. We validate TMN-Reweight at the 4B scale, where it markedly reduces cross-task advantage-scale inconsistency and raises the benchmark average from 62.2 to 63.0, with the largest gains on tasks requiring multi-source aggregation and reasoning. We will release the dataset, construction pipeline, training code, and both trained models upon acceptance.

1 Introduction

As large language models (LLMs) are increasingly deployed in document analysis, deep research, retrieval-augmented generation, and agentic workflows, long-context reasoning has become an important capability for advanced AI systems (Achiam et al., 2023; Bai et al., 2024). Although pretraining and mid-training have rapidly expanded model context windows (Yang et al., 2025; DeepSeek-AI, 2025; Chu et al., 2026), effectively using these extended contexts during post-training remains a key bottleneck. Recent studies show that reinforcement learning with verifiable rewards (RLVR) can further improve long-context utilization (Guo et al., 2025; Wang et al., 2026; Chen et al., 2026a). Together, these advances make post-training data design and optimization central to translating longer context windows into reliable gains across practical long-context workloads.

Existing long-context RL methods commonly construct training data by increasing retrieval complexity, for example through longer retrieval chains, more documents, additional distractors, or harder-to-find evidence (Wang et al., 2026; Chen et al., 2026a; Shen et al., 2025). While these designs provide useful supervision, their task interfaces remain centered on locating evidence and answering questions. Practical long-context applications also require cross-document aggregation, numerical reasoning, structured extraction, relevance ranking, sequence ordering, and long-document summarization. Training data organized primarily around retrieval paths consequently provides limited direct supervision for these capabilities. This gap is not merely a matter of dataset size:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

repeatedly increasing retrieval difficulty exercises a similar operation rather than broadening how models use evidence once it has been found.

The diversity of these tasks also calls for rewards that match their evaluation semantics. Precise retrieval can be evaluated by Exact Match, exhaustive retrieval by token-level F1, ranking by graded relevance, structured extraction by set overlap, and summarization by content coverage. Reducing all of them to binary question answering discards partial correctness, relative order, and output structure. Effective long-context RLVR data construction should therefore jointly organize the capability exercised by each task and the verifiable reward used to supervise it.

Following this design, we organize long-context training into nine task types based on the capability dimensions of LongBench Pro (Chen et al., 2026b): precise retrieval, evidence integration, exhaustive retrieval, numerical reasoning, structured extraction, structured matching, relevance ranking, sequence ordering, and summarization. We pair these tasks with EM, Accuracy, F1, `math_verify`, IoU, SubEM, NDCG, Pairwise, and ROUGE-L rewards, respectively. We then construct 22,965 long-context RLVR samples through two complementary tracks (Section 3). The curated track selects samples from annotated open-source corpora and converts their annotations into task-compatible reward formats. The synthetic track generates question–answer pairs from real books, papers, and multi-turn dialogues while retaining the original documents as context. Generated samples undergo full-context QA verification and multi-scale model verification to check question definition, answer support, distractor quality, and label reliability. Together, the two tracks combine the reliability of existing annotations with the breadth needed to cover underrepresented long-context abilities.

Training this heterogeneous mixture introduces an additional optimization challenge. The task-specific rewards differ in variance and density, and the per-prompt normalization in GRPO (Shao et al., 2024) can produce inconsistent advantage scales across tasks. Motivated by this training behavior and its gradient analysis, we propose **TMN-Reweight** (Section 4). Task-level mean normalization uses the root mean square of within-prompt standard deviations to align scales across tasks, while difficulty-adaptive reweighting adjusts positive and negative advantages according to prompt difficulty. The two mechanisms separate cross-task calibration from within-task difficulty correction, allowing each to address a distinct source of gradient imbalance.

Based on the capability-oriented dataset and multitask optimization, we build GoLongRL. First, we train with GRPO on the complete 23K dataset at two model scales. On Qwen3-4B-Thinking-2507, the average over six long-context benchmarks increases from 53.0 to 62.2; on Qwen3-30B-A3B-Thinking-2507, it increases from 60.1 to 69.8. At both scales, these results exceed those obtained by GRPO trained on data constructed by QwenLong-L1.5 (Shen et al., 2025). The trained models also show robust performance on general reasoning, agentic memory, dialogue memory, and overlong-context evaluations. In addition, we validate TMN-Reweight at the 4B scale. It substantially improves the consistency of advantage scales across tasks and raises the benchmark average from 62.2 to 63.0, with the largest gains on tasks requiring multi-source aggregation and reasoning. The dataset, construction pipeline, training code, and both models will be released upon acceptance. Our contributions are:

- **Capability-oriented data construction.** We construct a 23K-sample long-context RLVR dataset spanning nine task types, each paired with a task-aligned verifiable reward, through curated and real-document-grounded synthetic data. This construction couples broader task coverage with reward signals faithful to each output structure.
- **Cross-scale empirical evidence.** With GRPO, the 4B and 30B-A3B models improve by 9.2 and 9.7 points, respectively, over six long-context benchmarks. The similar gains across scales show that the benefit persists beyond a single model regime. We will release the dataset, construction pipeline, training code, and both trained models upon acceptance.
- **Heterogeneous multitask optimization.** We introduce TMN-Reweight to align cross-task advantage scales and account for prompt difficulty. We characterize its scale-alignment behavior and downstream effect at 4B, connecting the optimization mechanism to the observed multitask gains.

2 Related Work

Long-Context RL Training. Several recent methods demonstrate that RL post-training can improve long-context utilization beyond supervised fine-tuning. LoongRL (Wang et al., 2026) composes short multi-hop QA pairs into long-context tasks via KeyChain, inducing reasoning patterns that

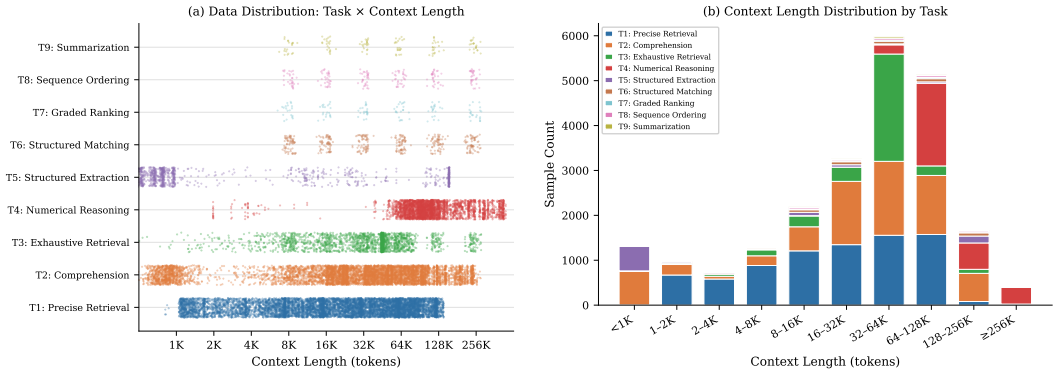


Figure 1: Dataset distribution across tasks and context lengths. The left panel shows the distribution of task types and lengths. The right panel reports sample counts within $\log_2$-scale length bins.

transfer from 16K training to 128K evaluation. LongRLVR (Chen et al., 2026a) identifies exponential gradient attenuation from sparse final-answer rewards and introduces dense context evidence signals to mitigate it. QwenLong-L1.5 (Shen et al., 2025) achieves strong results by combining atomic-fact-based data synthesis with task-specific advantage estimation (AEPO). Despite their progress, their data construction remains centered on retrieval-intensive QA, leaving output structures such as summarization and ranking with limited direct supervision. Following the capability dimensions of LongBench Pro (Chen et al., 2026b), our work instantiates nine task types as RLVR training samples and pairs each with a reward that preserves its output semantics.

GRPO Variants for Multitask RL. GRPO (Shao et al., 2024) has become the standard RLVR algorithm, computing advantages by per-prompt z-score normalization. Several variants modify different parts of this estimation: Dr. GRPO (Liu et al., 2025) removes the per-prompt standard deviation to reduce difficulty-dependent scaling; F-GRPO (Plyusov et al., 2026) applies difficulty-aware focal reweighting; HA-DW (Yang et al., 2026) uses historical reward statistics for adaptive weighting; and QwenLong-L1.5 (Shen et al., 2025) normalizes advantages within task types to reduce scale differences. These methods operate on prompt-, history-, or task-level statistics. TMN-Reweight combines task-level scale normalization with prompt-level difficulty reweighting, separating the two operations in heterogeneous multitask training.

3 Capability-Oriented Data Construction

Effective long-context data construction should start from the question of *what capabilities* long-context understanding requires, rather than from *how to inject length* into a prompt. This section describes the dataset composition (§3.1) and construction pipeline (§3.2). The construction follows three principles. First, *capability orientation* organizes each sample into one of nine task types derived from the capability dimensions of LongBench Pro (Chen et al., 2026b). Second, *reward alignment with task semantics* preserves the evaluation signal associated with each output structure rather than reducing all tasks to a uniform binary reward. Third, *real document grounding* retains authentic source texts as context and limits synthesis to the corresponding question–answer pairs.

3.1 Overview of the Dataset

The resulting dataset contains 22,965 samples with context lengths ranging from 0.1K to 256K tokens (Figure 1). Its empirical task distribution is non-uniform. T1 through T4 contribute 21,248 samples, or 92.5% of the dataset, reflecting the greater availability of high-quality real-document sources and scalable verification procedures for these formats. T5 through T9 represent the long tail of long-context capabilities, for which source material is scarcer and outputs are more costly to construct and verify. They contribute the remaining 1,717 samples, or 7.5%. Together, the nine task types expose the model to distinct output structures while retaining samples that satisfy the source and verification requirements of each task.

Table 1: Composition of the capability-oriented RLVR dataset. Each task is paired with a reward aligned with its output structure. The full source-to-task mapping is in Appendix B.

Task	Reward	#Samples	%	Scoring Semantics	Capability	Representative Sources
T1	EM	7,908	34.4	Exact match	Precise retrieval	CLongEval, WikiHop, NIAH (synth.)
T2	Accuracy	6,808	29.6	MC accuracy	Evidence integration	Gutenberg, arXiv, PMC, BEAM (synth.)
T3	F1	3,478	15.1	Token-level F1	Exhaustive retrieval	LongBench Pro, CAIL2018
T4	`math_verify`	3,054	13.3	Math verification	Numerical reasoning	FinancialQA
T5	IoU	937	4.1	Structured overlap	Structured extraction	MultiTableQA
T6	SubEM	360	1.6	Substring match	Structured matching	LongBench Pro
T7	NDCG	120	0.5	Graded relevance	Relevance ranking	LongBench Pro
T8	Pairwise	180	0.8	Pairwise comparison	Sequence ordering	LongBench Pro
T9	ROUGE-L	120	0.5	Content overlap	Summarization	LongBench Pro
	Total	**22,965**	**100**			

The reward interfaces preserve different forms of task-level correctness. EM, Accuracy, `math_verify`, and SubEM evaluate exact or rule-based correctness, while F1, IoU, NDCG, Pairwise, and ROUGE-L retain graded information about overlap, relevance, order, or content coverage. Each reward therefore expresses quality within its associated task, and the resulting differences in advantage scale motivate the multitask optimization developed in Section 4. Appendix Figure 5 provides a complementary embedding-space view of the task-associated structure in the dataset.

The dataset is assembled through two complementary construction tracks. The *curated track* draws roughly 14K samples from established long-context corpora with existing annotations, including CLongEval (Qiu et al., 2024), WikiHop (Welbl et al., 2018), LongBench Pro (Chen et al., 2026b), FinancialQA (Chen et al., 2021), CAIL2018 (Xiao et al., 2018), MultiTableQA (Wang et al., 2025), Gaokao Chinese (Zhang et al., 2024), and LexSum (Shen et al., 2022). The *synthetic track* generates QA pairs from real source documents, contributing roughly 9K samples from Project Gutenberg books (Lahiri, 2014), arXiv CC0 papers (M8than, 2024), PMC Open Access articles (National Library of Medicine, 2003), and multi-turn dialogues from BEAM (Tavakoli et al., 2026) and Oolong (Bertsch et al., 2025). The synthetic track targets T1 and T2 because their binary rewards enable reliable automatic verification.

Among the synthetic portion, T2 is the primary target for large-scale synthesis because its multiple-choice format supports both reliable generation and verification. To capture diverse reasoning patterns within T2, we construct three complementary categories. *Evidence integration QA* requires locating and synthesizing evidence from multiple scattered passages across long documents. *Multiple-choice rule induction* asks the model to infer an underlying rule from examples embedded in task-irrelevant contexts and select its correct application to a new case, with Accuracy as the reward. *Dialogue memory and tracking* uses multi-turn dialogues exceeding 50 turns and 30K tokens, targeting information stated by a specific speaker in an earlier turn. We additionally synthesize approximately 4K needle-in-a-haystack samples for T1 to provide controlled examples of precise long-range retrieval.

3.2 Construction Pipeline

Both the curated and synthetic tracks are produced through a unified four-phase pipeline (Figure 2).

P1. Source corpus collection. Guided by the nine task types derived from LongBench Pro, this phase collects annotated datasets for the curated track and unannotated real-world documents for the synthetic track, selecting corpora to maximize diversity in domain, document structure, and length distribution.

P2. Task-oriented filtering and assignment. Each sample or document is assigned to exactly one task based on task-specific selection criteria. For curated datasets, we select subsets whose evaluation semantics align with a particular task. For unannotated documents, filtering operates at the document level based on structural properties required by each task.

P3. Sample construction. Filtered materials are converted into training-ready RLVR samples through two parallel tracks. *Curated samples* undergo compatibility filtering and reward format standardization, since existing annotations are not always directly convertible into reward-computable answers. *Synthetic samples* are produced in two steps. First, we utilize a synthesis model to construct

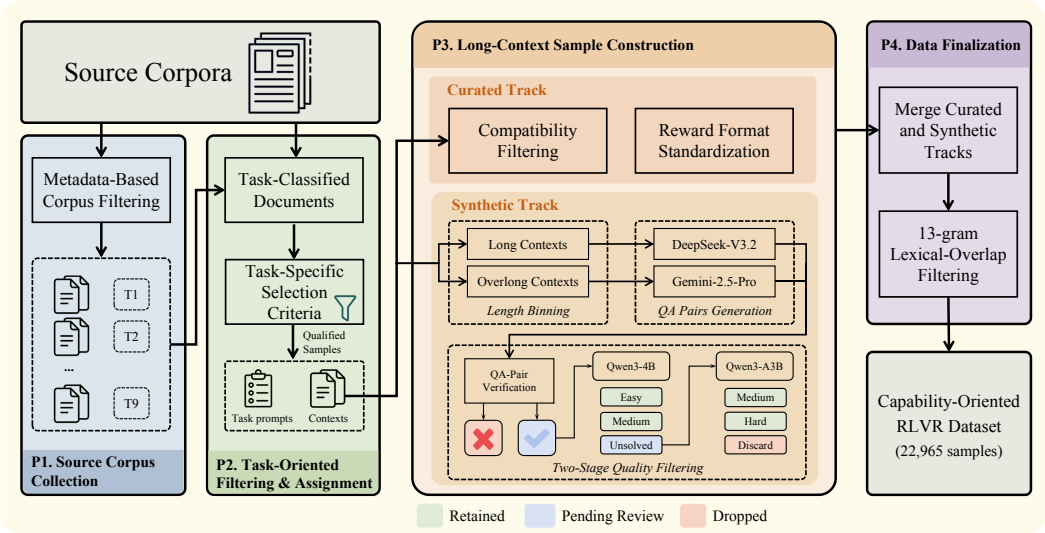


Figure 2: Overview of the four-phase capability-oriented RLVR dataset construction pipeline. Source corpora are first filtered and assigned to one of nine task types. Curated samples are converted through compatibility filtering and reward-format standardization, while synthetic samples are generated from length-binned real documents and undergo QA-pair and multi-scale model verification. The two tracks are then merged and filtered for 13-gram lexical overlap with evaluation queries, yielding 22,965 training samples.

QA pairs from long-context documents. To reduce API cost, documents are first tokenized and assigned to length bins, then DeepSeek-V3.2 (DeepSeek-AI, 2025) generates candidate QA pairs for documents within 128K tokens while Gemini-2.5-Pro (Comanici et al., 2025) handles overlong ones. Second, the generated QA pairs then undergo two-stage quality filtering. The first stage uses QA-pair verification by Gemini-2.5-Pro to confirm answer correctness and reject hallucinated or ambiguous labels. The second stage applies multi-scale verification by evaluating each sample with Qwen3-4B-Thinking-2507 and Qwen3-30B-A3B-Thinking-2507. Samples that neither model solves are removed as a conservative quality-control step (details in Appendix E).

P4. Dataset finalization. Samples from the curated and synthetic tracks are first merged. We then apply 13-gram lexical-overlap filtering between the resulting pool and all benchmark queries used in Section 5, removing any training sample that shares a 13-gram substring with an evaluation query. This step reduces the risk of direct lexical information leakage between training and evaluation. The retained samples form the final capability-oriented dataset of 22,965 training instances.

4 TMN-Reweight for Multitask Long-Context RL

The reward diversity that enables broad capability coverage also introduces an optimization challenge. Our nine reward functions, spanning exact match, token-level F1, NDCG, etc. differ in numerical scale, variance, and density, causing inconsistent advantage scales and imbalanced gradient contributions across tasks under standard advantage estimation. This section analyzes the source of this imbalance (§4.1) and presents TMN-Reweight, which addresses it via task-level mean normalization (§4.2) and difficulty-adaptive reweighting (§4.3).

4.1 Background and Problem Analysis

GRPO (Shao et al., 2024) has become the dominant algorithm for RLVR, and most subsequent improvements, including DAPO (Yu et al., 2026) and GPPO (Su et al., 2026a;b), build on its framework. We therefore adopt GRPO as our base algorithm throughout. For each prompt u, GRPO samples G candidate responses from the current policy and computes the response advantage through

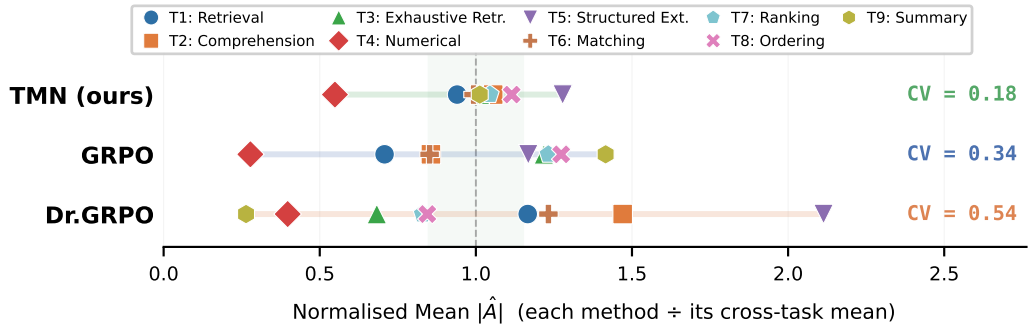


Figure 3: Normalized mean absolute advantage per task. Consistent with the gradient analysis, TMN reduces the cross-task coefficient of variation from 0.34 under GRPO to 0.18 and aligns the nine task-level advantage scales around their shared mean.

group-level z-score normalization.

$$A_i^u = \frac{r_i - \mu_u}{\sigma_u + \delta}, \quad \mu_u = \frac{1}{G}\sum_{j=1}^{G} r_j$$
$$\sigma_u = \sqrt{\frac{1}{G-1}\sum_{j=1}^{G}(r_j - \mu_u)^2} \tag{1}$$

The policy is then updated by maximizing a clipped surrogate objective. Throughout this paper, δ denotes a small constant (10^{-8}) for numerical stability.

However, per-prompt normalization in Eq. 1 introduces two biases under heterogeneous multitask training. First, extreme-difficulty prompts have small σ_u, which assigns disproportionately large normalized advantages to their rare outcomes while comparatively compressing medium-difficulty prompts with more diverse rollouts (Liu et al., 2025; Yang et al., 2026) (*difficulty bias*). Dr. GRPO removes σ_u to concentrate gradients on more informative samples, but this exposes a second issue: tasks with higher reward variance (e.g., F1-based retrieval vs. binary accuracy) then generate disproportionately large gradients and dominate optimization (*scale inconsistency*). The two issues call for opposing interventions; TMN-Reweight resolves this tension through two sequential steps.

4.2 Step 1. Task-Level Mean Normalization

Instead of normalizing by the per-prompt σ_u, we normalize by a task-level aggregate computed as the root mean square of per-prompt standard deviations within each task:

$$\hat{A}_i^u = \frac{r_i - \mu_u}{\sigma_{\text{task}(u)} + \delta} \quad \sigma_{\text{task}(u)} = \sqrt{\frac{1}{|U_{\text{task}}|}\sum_{u \in U_{\text{task}}} \sigma_u^2} \tag{2}$$

This formulation is motivated by a gradient norm analysis (Appendix A) showing that the per-task gradient magnitude scales with $\sqrt{\mathbb{E}_{u\sim\mathcal{T}_k}[\sigma_u^2]}$, of which $\sigma_{\text{task}(u)}$ is the empirical estimate computed over all prompts of the same task within the current training batch. Because the denominator is shared across all prompts in the same task, this operation aligns gradient scales across tasks while preserving relative advantage magnitudes within each task.

Figure 3 examines whether task-level mean normalization produces the scale-alignment behavior predicted by the gradient analysis. We compute the mean absolute advantage per task under GRPO, Dr. GRPO, and TMN over the full training set, then normalize each method by its cross-task mean, with 1.0 denoting the shared mean. Dr. GRPO yields a coefficient of variation (CV) of 0.54 because removing σ_u retains the differences induced by task-level reward variance. GRPO reduces the CV to 0.34 through per-prompt normalization, while cross-task dispersion remains because σ_u operates within each prompt. TMN further reduces the CV to 0.18, confirming the intended reduction in advantage-scale inconsistency across the nine tasks.

4.3 Step 2. Difficulty-Adaptive Reweighting

Task-level mean normalization aligns gradient scales across tasks, but difficulty bias within each task remains. A prompt the model already solves reliably contributes the same gradient magnitude as one where only a single rollout out of G succeeds, even though the latter represents a more informative learning opportunity. To allocate gradient budget according to where the model can most benefit, we introduce difficulty-adaptive reweighting.

We estimate prompt difficulty via a smoothed relative-success rate $\hat{p} = \frac{1}{G}\sum_{i=1}^{G} \mathbf{1}[r_i > \tilde{\mu}_u]$, where $\tilde{\mu}_u = \alpha \cdot \mu_u + (1-\alpha) \cdot \mu_{\text{task}}$ interpolates between the prompt-level mean μ_u and the task-level batch mean $\mu_{\text{task}} = \frac{1}{|U_{\text{task}}|}\sum_{u' \in U_{\text{task}}} \mu_{u'}$ to reduce estimation variance from small group sizes. This task-local comparison maps heterogeneous reward values to a common indicator of whether each rollout exceeds its smoothed baseline. The difficulty weight $w = \exp(0.5 - \hat{p})$ amplifies hard prompts ($\hat{p} < 0.5$, $w > 1$) and downweights easy ones ($\hat{p} > 0.5$, $w < 1$), while centering the exponential at 0.5 gives $w = 1$ at $\hat{p} = 0.5$.

Difficulty-adaptive reweighting uses prompt difficulty to determine which *side* of the advantage distribution receives emphasis. We apply w asymmetrically according to the sign of the TMN advantage $\hat{A}_i^u$, concentrating each update on the trajectories that provide the most useful learning signal at the corresponding difficulty level. Because $\hat{A}_i^u$ uses the original group mean μ_u as its baseline (Eq. 2), standard reinforcement semantics are preserved, while the smoothed $\tilde{\mu}_u$ enters only through the relative-success rate $\hat{p}$ for difficulty estimation.

$$\tilde{A}_i = \begin{cases} \hat{A}_i^u \cdot w & \text{if } \hat{A}_i^u > 0 \\ \hat{A}_i^u \cdot \dfrac{1}{w} & \text{otherwise} \end{cases} \tag{3}$$

This formulation creates four gradient regimes. For hard prompts ($w > 1$), positive advantages amplify the contribution of relatively rare higher-reward trajectories, while negative advantages are scaled down by $1/w$. For easy prompts ($w < 1$), positive advantages are downweighted to reduce redundant updates from already strong trajectories, while negative advantages amplify the corrective signal from unexpected lower-reward outcomes.

5 Experiments

In this section, we evaluate whether capability-oriented data improves long-context performance across model scales (Section 5.1) and whether the resulting gains extend to overlong contexts, general reasoning, and memory. We then hold the data and 4B backbone fixed to examine how TMN-Reweight affects heterogeneous multitask optimization (Section 5.2).

Training. We train Qwen3-4B-Thinking-2507 and Qwen3-30B-A3B-Thinking-2507 on the complete 22,965-instance dataset constructed in Section 3. The cross-scale experiments use GRPO, while the TMN-Reweight comparison reuses the 4B backbone and training data so that only the optimization method changes. We conduct all runs with `verl` in a strictly on-policy setting, using a batch size of 128, a group size of 16, and a learning rate of 2×10^{-6} with 5 warmup steps. Temperature and top-p are set to 1.0 during rollout generation. Appendix C provides the complete configuration.

Evaluation. We follow the evaluation protocol of QwenLong-L1.5 (Shen et al., 2025) on six long-context benchmarks up to 128K tokens, with protocol alignment verified in Appendix D. These benchmarks comprise LongBench V2 (Bai et al., 2025), MRCR (Vodrahalli et al., 2024), Frames (Krishna et al., 2025), LongBench V1 QA subsets (Bai et al., 2024), DocMath (Zhao et al., 2024), and CorpusQA (Lu et al., 2026). We additionally assess extrapolation on MRCR and CorpusQA beyond 128K tokens, general reasoning on MMLU-Pro (Wang et al., 2024), AIME24/25 (Mathematical Association of America, 2025), and GPQA-Diamond (Rein et al., 2023), as well as agentic memory (Patil et al., 2025) and dialogue memory (Wu et al., 2024).

5.1 Capability-Oriented Data Evaluation

Long-context capability. Table 2 evaluates the central data hypothesis under GRPO. Relative to the base models, the six-benchmark average rises by 9.2 points at 4B and 9.7 points at 30B. Similar

Table 2: Long-context results up to 128K tokens. Within each model group, the best result is **bolded** and the second best is underlined.

Model	DocMath	LongBench V2	Frames	MRCR	CorpusQA	LongBench V1	Average
Frontier Models							
GPT-5†	67.6	62.8	**84.6**	77.3	**81.6**	**73.7**	74.7
Gemini-2.5-Pro†	62.4	**65.7**	74.5	79.9	80.6	71.3	72.4
Qwen3-Max-Thinking-Preview†	64.1	57.9	77.9	71.2	74.7	70.7	69.4
DeepSeek-R1-0528‡	63.4	59.5	76.9	64.9	77.5	69.9	68.7
Qwen3-235B-A22B-Thinking-2507‡	65.8	57.5	75.1	66.2	75.3	70.9	68.5
Gemini-2.5-Flash-Thinking†	64.8	56.8	65.8	78.8	79.4	66.9	68.7
GPT-OSS-120B†	61.3	47.0	72.7	39.7	64.4	66.3	58.6
GPT-5-Nano†	63.9	43.7	73.5	43.9	50.3	67.1	57.1
Qwen3.5-4B‡	65.9	45.9	70.9	96.1	70.5	63.6	68.8
Qwen3.5-35B-A3B‡	**68.9**	59.4	74.9	**97.4**	78.7	70.2	**74.9**
4B Models							
Qwen3-4B-Thinking-2507	61.0	40.2	64.4	38.4	49.9	64.0	53.0
QwenLong-L1.5-4B (w. GRPO)†	61.3	44.3	**67.1**	40.9	58.8	64.1	56.1
GoLongRL-4B (w. GRPO)	**62.5**	**45.5**	66.6	**67.5**	**65.1**	**65.9**	**62.2**
30B Models							
Qwen3-30B-A3B-Thinking-2507	63.3	48.7	70.2	41.6	70.5	66.5	60.1
QwenLong-L1.5-30B-A3B (w. GRPO)†	65.1	**55.3**	71.4	66.9	**76.9**	67.9	67.2
GoLongRL-30B-A3B (w. GRPO)	**65.3**	55.1	**74.5**	**81.6**	73.6	**68.7**	**69.8**

† The two QwenLong-L1.5 GRPO checkpoints and their training data have not been publicly released and therefore cannot be re-evaluated. Their results, together with those of the other models marked by †, are taken from QwenLong-L1.5 (Shen et al., 2025) to preserve a consistent evaluation setup. We validate the alignment between our evaluation setup and that of QwenLong-L1.5 in Appendix D.
‡ We evaluate these publicly released models using our unified evaluation setup.

Table 3: Overlong-context results beyond 128K tokens. Best and second-best results within each scale are **bolded** and underlined. The QwenLong-L1.5 GRPO checkpoints are not publicly available, so we cannot evaluate them on these benchmarks.

Model	MRCR 128K–512K	MRCR 512K–1M	CorpusQA 1M	Average
Qwen3-4B-Thinking-2507	11.2	1.4	**5.8**	6.1
GoLongRL-4B (w. GRPO)	**23.4**	**4.9**	5.5	**11.3**
Qwen3-30B-A3B-Thinking-2507	24.9	3.7	6.4	11.7
GoLongRL-30B-A3B (w. GRPO)	**37.5**	**9.1**	**9.1**	**18.6**

gains from different starting performance show that capability supervision remains useful as model capacity increases. These benchmarks exercise distinct operations, including numerical reasoning, multi-hop inference, and corpus-level evidence integration. Improvements across them show that the benefit extends beyond a single way of using retrieved evidence. GoLongRL also exceeds the reported QwenLong-L1.5 GRPO average at both scales. These results across tasks and scales supports capability coverage as a complementary axis to retrieval difficulty in long-context data construction.

Longer contexts. Table 3 tests transfer beyond the training range. At 512K–1M tokens, MRCR improves by 3.5 points at 4B and 5.4 points at 30B, showing that evidence recovery transfers to unseen lengths. CorpusQA at 1M remains near the base score at 4B but improves at 30B, suggesting that corpus-level synthesis imposes an additional capacity demand. This contrast motivates evaluating context length jointly with the required evidence operation.

Reasoning and memory. Table 4 tests transfer beyond long-document QA. All four reasoning benchmarks improve at both scales despite being absent from training, showing that specialization preserves broader reasoning. Memory transfer differs by operation. Gains are largest for recursive summarization (+9.7 at 4B and +4.5 at 30B) and LongMemEval (+13.6 at both scales), whereas direct key–value lookup declines slightly at 4B. This pattern links transfer to integrating and reconstructing stored evidence rather than to every form of memory access.

Table 4: General reasoning and memory results under GRPO. Arrows show changes from the base model; **bold** and underlined mark the best and second-best results within each scale. The QwenLong-L1.5 GRPO checkpoints are not publicly available, so we cannot evaluate them on these benchmarks.

Model	General Reasoning				Agentic Memory			Dialogue Memory
	MMLU-Pro	AIME24	AIME25	GPQA	Memory-KV	Memory-Vec	Memory-Rec_Sum	LongMemEval
4B Models								
Qwen3-4B-Thinking-2507	73.6	82.6	80.2	65.4	**12.9**	15.5	36.8	47.6
GoLongRL-4B (w. GRPO)	**74.3**$^{+0.7}$	**84.2**$^{+1.6}$	**80.8**$^{+0.6}$	**67.7**$^{+2.3}$	11.6$^{-1.3}$	**20.0**$^{+4.5}$	**46.5**$^{+9.7}$	**61.2**$^{+13.6}$
30B Models								
Qwen3-30B-A3B-Thinking-2507	80.2	90.5	85.1	70.1	15.5	18.7	31.0	61.6
GoLongRL-30B-A3B (w. GRPO)	**81.0**$^{+0.8}$	**91.3**$^{+0.8}$	**86.9**$^{+1.8}$	**72.3**$^{+2.2}$	**16.1**$^{+0.6}$	**21.9**$^{+3.2}$	**35.5**$^{+4.5}$	**75.2**$^{+13.6}$

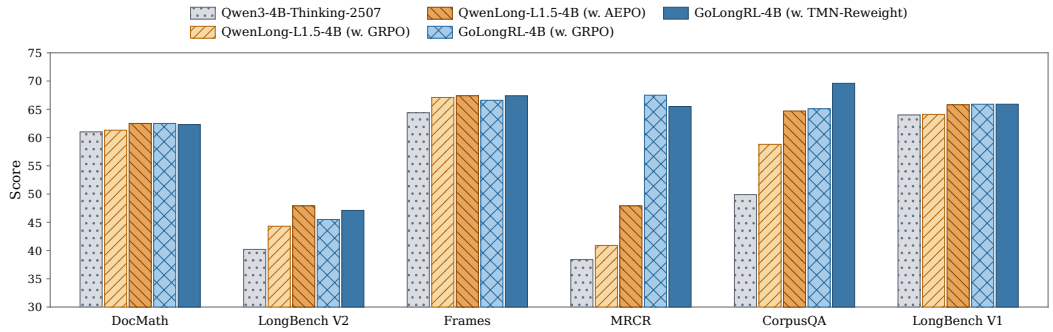


Figure 4: Six-benchmark results of five 4B configurations. QwenLong-L1.5 results are from Shen et al. (2025) because its checkpoints and data are unavailable.

5.2 TMN-Reweight Validation

Having established the contribution of capability-oriented data under GRPO, we fix the training data and 4B backbone to study optimization under heterogeneous rewards. We use $\alpha = 0.8$ for $w = \exp(0.5 - \hat{p})$ in this experiment and report its sensitivity analysis in Appendix C.1. Figure 4 shows that TMN-Reweight raises the six-benchmark average from 62.2 to 63.0. CorpusQA gains 4.5 points and LongBench V2 gains 1.6, while MRCR decreases by 2.0. The changes are not uniform across benchmarks, and the overall benefit is concentrated on tasks that require aggregating information from multiple sources and reasoning over it. This pattern shows that calibrating the optimization contributions of heterogeneous rewards can improve multitask learning beyond the gains already supplied by the data, particularly for capabilities that must coordinate multiple pieces of evidence. Capability-oriented data determine which operations receive supervision, whereas TMN-Reweight mitigates the update imbalance induced by different reward scales, allowing these capabilities to be optimized more coherently within a single training process.

6 Conclusion

We present GoLongRL, a capability-oriented framework that pairs nine long-context task types with rewards faithful to their output semantics. Training on the resulting 23K dataset with GRPO improves both model scales and transfers to reasoning, memory, and longer contexts. Across these evaluations, the strongest transfer emerges when models must integrate or transform evidence, suggesting that capability diversity develops reusable operations beyond retrieval. At 4B, the proposed TMN-Reweight further aligns advantage scales and strengthens multi-source aggregation. Together, the results support jointly designing capability coverage and heterogeneous learning signals for long-context RLVR.

AI USE STATEMENT

In this work, we used LLMs to generate and verify synthetic question–answer pairs as part of the data construction procedure described in Section 3.2. All generated training samples passed the quality-control stages described in Section 3.2. The authors reviewed all AI-assisted content and take responsibility for the final text, claims, and artifacts.

ETHICS STATEMENT

This work uses existing public datasets and publicly accessible documents and does not recruit human participants or collect new user data. Appendix B documents the source datasets used in construction. Because public documents may contain copyrighted, biased, or sensitive content, users of the released dataset and models should follow the licenses and usage conditions of the original sources and assess risks for their intended applications.

REPRODUCIBILITY STATEMENT

Section 3 describes the dataset composition and construction pipeline, and Appendix B provides the source-to-task mapping. Section 4 and Appendix A specify TMN-Reweight and its theoretical motivation. Appendix C reports the training configuration and sensitivity analysis, Appendix D documents the evaluation protocol, and Appendix E provides the generation and verification prompts. We will release the dataset, construction pipeline, training code, and both trained models upon acceptance.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. Longbench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 3119–3137, 2024.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, et al. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664, 2025.

Amanda Bertsch, Adithya Pratapa, Teruko Mitamura, Graham Neubig, and Matthew R. Gormley. Oolong: Evaluating long context reasoning and aggregation capabilities, 2025.

Guanzheng Chen, Michael Qizhe Shieh, and Lidong Bing. LongRLVR: Long-context reinforcement learning requires verifiable context rewards. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3697–3711. Association for Computational Linguistics, 2021.

Ziyang Chen, Xing Wu, Junlong Jia, Chaochen Gao, Qi Fu, Debing Zhang, and Songlin Hu. Longbench pro: A more realistic and comprehensive bilingual long-context evaluation benchmark, 2026b.

Chenglong Chu, Guorui Zhou, Guowang Zhang, Han Li, Hao Peng, Hongtao Cheng, Jian Liang, Jiangxia Cao, Kun Gai, Lingzhi Zhou, et al. Kwai summary attention technical report. *arXiv preprint arXiv:2604.24432*, 2026.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025.

DeepSeek-AI. Deepseek-v3.2: Pushing the frontier of open large language models, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. 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)*, pp. 4745–4759, 2025.

Shibamouli Lahiri. Complexity of Word Collocation Networks: A Preliminary Structural Analysis. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 96–105. Association for Computational Linguistics, April 2014.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective, 2025.

Zhiyuan Lu, Chenliang Li, Yingcheng Shi, Weizhou Shen, Ming Yan, and Fei Huang. Corpusqa: A 10 million token benchmark for corpus-level analysis and reasoning. *arXiv preprint arXiv:2601.14952*, 2026.

recursal.ai M8than. arxiv-cc0-v0.5. https://huggingface.co/datasets/recursal/arXiv-CC0-v0.5, 2024.

Mathematical Association of America. American invitational mathematics examination (aime). https://maa.org/maa-invitational-competitions/, 2025.

National Library of Medicine. PMC Open Access Subset. https://pmc.ncbi.nlm.nih.gov/tools/openftlist/, 2003.

Shishir G Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E Gonzalez. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*, 2025.

Daniil Plyusov, Alexey Gorbatovski, Boris Shaposhnikov, Viacheslav Sinii, Alexey Malakhov, and Daniil Gavrilov. F-grpo: Don't let your policy learn the obvious and forget the rare, 2026.

Zexuan Qiu, Jingjing Li, Shijue Huang, Xiaoqi Jiao, Wanjun Zhong, and Irwin King. CLongEval: A Chinese benchmark for evaluating long-context large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, 2024.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. GPQA: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*, 2023.

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

Weizhou Shen, Ziyi Yang, Chenliang Li, Zhiyuan Lu, Miao Peng, Huashan Sun, Yingcheng Shi, Shengyi Liao, Shaopeng Lai, Bo Zhang, Dayiheng Liu, Fei Huang, Jingren Zhou, and Ming Yan. Qwenlong-l1.5: Post-training recipe for long-context reasoning and memory management, 2025.

Zejiang Shen, Kyle Lo, Lauren Yu, Nathan Dahlberg, Margo Schlanger, and Doug Downey. Multilexsum: Real-world summaries of civil rights lawsuits at multiple granularities. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

Zhenpeng Su, Leiyu Pan, Xue Bai, Dening Liu, Guanting Dong, Jiaming Huang, Minxuan Lv, Wenping Hu, Fuzheng Zhang, Kun Gai, and Guorui Zhou. Klear-reasoner: Advancing reasoning capability via gradient-preserving clipping policy optimization, 2026a.

Zhenpeng Su, Leiyu Pan, Minxuan Lv, Yuntao Li, Wenping Hu, Fuzheng Zhang, Kun Gai, and Guorui Zhou. Ce-gppo: Coordinating entropy via gradient-preserving clipping policy optimization in reinforcement learning, 2026b. URL https://arxiv.org/abs/2509.20712.

Mohammad Tavakoli, Alireza Salemi, Carrie Ye, Mohamed Abdalla, Hamed Zamani, and J Ross Mitchell. Beyond a million tokens: Benchmarking and enhancing long-term memory in LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026.

Kiran Vodrahalli, Santiago Ontanon, Nilesh Tripuraneni, Kelvin Xu, Sanil Jain, Rakesh Shivanna, Jeffrey Hui, Nishanth Dikkala, Mehran Kazemi, Bahare Fatemi, et al. Michelangelo: Long context evaluations beyond haystacks via latent structure queries. *arXiv preprint arXiv:2409.12640*, 2024.

Siyuan Wang, Gaokai Zhang, Li Lyna Zhang, Ning Shang, Fan Yang, Dongyao Chen, and Mao Yang. LoongRL: Reinforcement learning for advanced reasoning over long contexts. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xixi Wang, Miguel Costa, Jordanka Kovaceva, Shuai Wang, and Francisco C. Pereira. Plugging schema graph into multi-table QA: A human-guided framework for reducing LLM reliance. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 5829–5842, Suzhou, China, 2025. Association for Computational Linguistics.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-PRO: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302, 2018.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. *arXiv preprint arXiv:2410.10813*, 2024.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. Cail2018: A large-scale legal dataset for judgment prediction, 2018.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, et al. Qwen3 technical report, 2025.

Fengkai Yang, Zherui Chen, Xiaohan Wang, Xiaodong Lu, Jiajun Chai, Guojun Yin, Wei Lin, Shuai Ma, Fuzhen Zhuang, Deqing Wang, Yaodong Yang, Jianxin Li, and Yikun Ban. Your group-relative advantage is biased, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, et al. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. Evaluating the performance of large language models on gaokao benchmark, 2024.

Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. Docmath-eval: Evaluating math reasoning capabilities of llms in understanding long and specialized documents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16103–16120, 2024.

A Gradient Analysis and Theoretical Motivation for TMN

Section 4.2 introduces task-level mean normalization and reports its scale-alignment behavior. This section provides the corresponding gradient-based derivation. Starting from the REINFORCE-style policy gradient used by Dr. GRPO, we show that the expected within-prompt reward variance governs the per-task gradient magnitude and leads to the $\sigma_{\text{task}(u)}$ denominator in Eq. 2.

A.1 Per-Prompt Gradient Norm Bound

Consider a GRPO variant that removes σ_u, as in Dr. GRPO. The policy gradient takes the form

$$\nabla_\theta \mathcal{J} = \mathbb{E}_{u,\,o}\Big[\big(r(u,o) - \mu_u\big)\,\nabla_\theta \log \pi_\theta(o \mid u)\Big] \tag{4}$$

For a single prompt u, the Cauchy–Schwarz inequality gives

$$\big\|\nabla_\theta \mathcal{J}_u\big\|^2 \;\le\; \mathbb{E}_o\Big[\big(r(u,o) - \mu_u\big)^2\,\big\|\nabla_\theta \log \pi_\theta(o \mid u)\big\|^2\Big] \tag{5}$$

Assuming the log-probability gradient norm varies modestly compared with the reward deviation, the dominant factor controlling the per-prompt gradient magnitude is the within-prompt reward variance $\sigma_u^2 = \mathbb{E}_o[(r(u,o) - \mu_u)^2]$.

A.2 Per-Task Gradient Magnitude and the Normalization Constant

For K task types $\{\mathcal{T}_k\}_{k=1}^K$, the gradient contribution from task k is governed by the expected within-prompt variance

$$\mathbb{E}_{u\sim\mathcal{T}_k}\big[\sigma_u^2\big] = \mathbb{E}_{u\sim\mathcal{T}_k}\big[\mathrm{Var}(r^{(k)} \mid u)\big] \tag{6}$$

To equalize gradient magnitudes across tasks, we introduce a per-task scaling coefficient

$$c_k = \frac{1}{\sqrt{\mathbb{E}_{u\sim\mathcal{T}_k}[\sigma_u^2] + \delta}} \tag{7}$$

The empirical estimate of $\mathbb{E}_{u\sim\mathcal{T}_k}[\sigma_u^2]$ over $|U_{\text{task}}|$ prompts from task k is $\frac{1}{|U_{\text{task}}|}\sum_{u\in U_{\text{task}}}\sigma_u^2$, whose square root recovers $\sigma_{\text{task}(u)}$ as defined in Eq. 2.

The key distinction is that $\sigma_{\text{task}(u)}$ estimates the expected *within-prompt* reward variance, not the pooled total variance obtained by aggregating all reward samples across prompts. These two quantities differ because the pooled variance conflates within-prompt variation, which drives gradient magnitude, with between-prompt variation in mean reward, which reflects difficulty differences rather than scale differences. Computing $\sigma_{\text{task}(u)}$ as the root mean square of per-prompt σ_u values therefore yields a normalization constant that equalizes gradient scales without distorting the within-task difficulty structure that TMN-Reweight subsequently exploits in its reweighting step.

B Data Sources

Section 3.1 summarizes the nine task types, reward interfaces, and dataset distribution. This section provides the complete source-to-task mapping behind Table 1 and an embedding-space view of the resulting dataset. The synthetic track generates QA pairs from real documents for T1 and T2, while the curated track covers all nine task types through established long-context corpora with existing annotations.

Figure 5 complements the task counts and context-length distributions in Figure 1. The task-associated structure reflects differences in content, source corpora, and output formats without implying separability between capabilities.

C Training Hyperparameters

Section 5 summarizes the shared training setup used for the cross-scale GRPO experiments and the 4B TMN-Reweight comparison. This section reports the complete configuration. The GRPO hyperparameters are shared by Qwen3-4B-Thinking-2507 and Qwen3-30B-A3B-Thinking-2507, while the TMN-Reweight coefficient applies only to the 4B experiment. All runs use `verl` in a strictly on-policy setting without a KL penalty.

Table 5: Complete mapping from data sources to tasks and reward types. Sample counts and capability descriptions are provided in Table 1.

Origin	Source	Task	Reward
Synthetic	Evidence integration QA (Gutenberg, arXiv CC0, PMC)	T2	Accuracy
	Rule induction reasoning with MCQs	T2	Accuracy
	Dialogue memory and tracking (BEAM, Oolong)	T2	Accuracy
	Needle in a haystack samples	T1	EM
Curated	CLongEval (novels, news, tables)	T1	EM
	Filtered WikiHop	T1	EM
	Gaokao Chinese, LexSum, LongBench Pro	T2	Accuracy
	LongBench Pro, CAIL2018	T3	F1
	FinancialQA	T4	math_verify
	MultiTableQA	T5	IoU
	LongBench Pro (clustering, consistency, rule induction)	T6	SubEM
	LongBench Pro (ranking subset)	T7	NDCG
	LongBench Pro (ordering and frequency subsets)	T8	Pairwise
	LongBench Pro (summarization subset)	T9	ROUGE-L

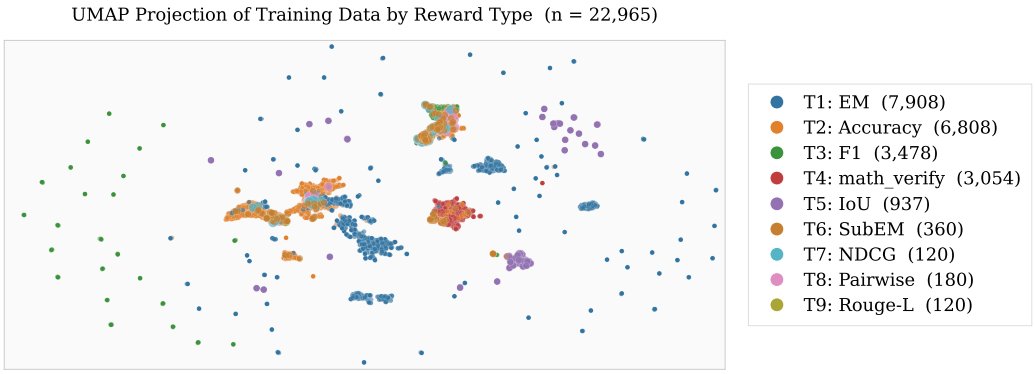


Figure 5: UMAP projection of the 22,965 training samples colored by task and reward interface. The visualization shows task-associated structure in the embedding space.

C.1 Sensitivity to the Reweighting Coefficient

Section 5.2 uses the pre-specified value $\alpha = 0.8$ for the 4B TMN-Reweight experiment. Table 7 compares it with alternative interpolation coefficients. The average varies smoothly across the tested settings and peaks at $\alpha = 0.8$, while the per-benchmark optima remain distributed across different values.

D Evaluation Protocol Alignment

Section 5 follows the QwenLong-L1.5 evaluation protocol and distinguishes reported results from models evaluated under our setup. This section quantifies the remaining cross-implementation differences on the six long-context benchmarks used in Table 2. Table 8 presents each published score with $\Delta = \text{ours} - \text{published}$ as a subscript. Gray marks $|\Delta| < 4$, while **orange** marks larger deviations. All models evaluated by us use identical generation settings. The deltas characterize comparability with reported results rather than asserting exact reproduction across evaluation implementations.

For the released QwenLong-L1.5-30B-A3B checkpoint, five of the six long-context scores differ by at most 0.5 points, with LongBench V1 differing by 2.7 points. The larger MRCR deviation for the Qwen3-30B-A3B-Thinking-2507 base model is retained in the table so that the cross-source comparison in Table 2 remains transparent.

Table 6: Training hyperparameters for GOLONGRL.

Hyperparameter	Value
Max prompt length	256K
Max response length	16K
Responses per prompt	16
Learning rate	2e-6
LR warmup steps	5
Weight decay	0.1
Gradient clipping	1.0
PPO epochs	1
Total epochs	10
Train prompt batch size	128
Clip ratio (low/high)	0.2 / 0.28
TMN-Reweight α	0.8

Table 7: Effect of the smoothing coefficient α in difficulty-adaptive reweighting at the 4B scale. Best results per column are **bolded**; second best are underlined.

α	Average	DocMath	LongBench V2	Frames	MRCR	CorpusQA	LongBench V1
0.0	61.3	61.9	45.8	**68.7**	64.5	60.2	**66.5**
0.5	62.9	**63.5**	**47.5**	67.4	64.9	67.8	66.2
0.8	**63.0**	62.3	47.1	67.4	**65.5**	**69.6**	65.9
1.0	61.5	62.4	44.9	66.8	65.4	65.7	63.8

E TASK INSTRUCTIONS AND DATA GENERATION PROMPTS

Section 3.2 summarizes sample construction, QA-pair verification, and reward-format standardization. This section documents the shared generation instruction and Gemini-2.5-Pro verification prompt used by the synthetic track, followed by the task-specific instructions and output contracts used to form training samples. The section headings follow the nine task names in Table 1, while the finer-grained labels inside each prompt preserve the corresponding source-task definitions.

We provide task-specific blocks for T2 through T9. T1 (Precise Retrieval) uses curated open-source samples and needle-in-a-haystack synthesis with the shared instruction directly.

E.1 SHARED GENERATION INSTRUCTION

Shared Sample Generation Instruction

```
You are a professional question-design specialist with capabilities in textual
deconstruction, deep reading, and cross-domain item construction.

Task: Given an input long context and its task category (primary task +
secondary task), create three questions that fulfill the task requirements,
including answers, design rationale, and detailed solution process.

Procedure:
1. Language Identification
   Determine the language of the input long text. All subsequent content
   must fully use the same language.

2. Question Construction
   - Generate three questions satisfying the secondary task specifications.
   - Questions must meet the evaluation goals of the primary task, the
     required I/O format, and the secondary task definition.
   - The three questions must exhibit clear differentiation, avoiding
     repeated formats or duplicate assessment points.
   - Identifier Rules: Use naturally occurring unique elements (titles,
     section names, entities) as identifiers.
   - For multiple-choice questions: include plausible distractors, at
```

Table 8: Protocol alignment on long-context benchmarks. Each cell shows the published score with Δ = ours − published.

Model	DocMath	LongBench V2	Frames	MRCR	CorpusQA	LongBench V1
Qwen3-4B-Thinking-2507	59.0 $_{+2.0}$	41.4 $_{-1.2}$	62.9 $_{+1.5}$	39.9 $_{-1.5}$	49.9 $_{\pm 0.0}$	64.3 $_{-0.3}$
Qwen3-30B-A3B-Thinking-2507	62.3 $_{+1.0}$	49.1 $_{-0.4}$	70.3 $_{-0.1}$	51.3 $_{-9.7}$	71.6 $_{-1.1}$	67.1 $_{-0.6}$
QwenLong-L1.5-30B-A3B	66.3 $_{+0.1}$	55.3 $_{-0.1}$	74.8 $_{-0.3}$	83.0 $_{-0.5}$	81.3 $_{-0.4}$	70.4 $_{-2.7}$

```
     least four options, and a text-verifiable correct option.
   - Provide design rationale and solution with evidence for every question.

3. Self-Check
   Verify task compliance, answer accuracy, absence of ambiguity, and
   language consistency. Regenerate if any criterion fails.

Input:
[Long Context]: {Long_Context}
[Primary Task]: {Primary_Task}
[Secondary Task]: {Secondary_Task}
```

E.2 QA-Pair Verification Prompt

QA-Pair Verification Prompt (Gemini-2.5-Pro)

```
You are a rigorous quality assessor for long-context question-answer pairs.
Given a source document and a generated QA sample, perform all of the
following checks.

1. Answer Uniqueness
   Verify that the correct answer is uniquely and unambiguously supported
   by the source document without requiring external knowledge.

2. Distractor Quality
   Confirm that each distractor is plausible but clearly wrong according
   to the source document.

3. Hallucination Detection
   Check whether the question, answer, or rationale contains any claim
   not grounded in the source document.

4. Task Compliance
   Verify that the question targets the intended reasoning capability and
   that the answer format matches the task specification.

5. Language and Format Consistency
   Confirm that all content is in the same language as the source document.

Output format:
Verdict -- PASS or REJECT
If REJECT, specify which check(s) failed and provide a brief explanation.

Input:
[Source Document]: {Source_Document}
[Question]: {Question}
[Options]: {Options}
[Correct Answer]: {Correct_Answer}
[Solution Rationale]: {Solution_Rationale}
```

E.3 T2: Evidence Integration

E.3.1 T2-a: Multi-Document Integration QA

T2-a: Multi-Document Integration QA

```
Primary Task: Evidence-Grounded QA
Answer fact/reasoning questions based on evidence.

Secondary Task: Multi-Doc Integration QA
Use multi-hop information across documents to answer questions.
Context Requirement: Full | Metric: Accuracy

Output format:
Output the "[Answer]" identifier first, then output the answer option
letter (A/B/C/D), without any additional content.
[Answer] C
```

E.3.2 T2-b: Single-Hop Fact QA

T2-b: Single-Hop Fact QA

```
Primary Task: Evidence-Grounded QA
Answer fact/reasoning questions based on evidence.

Secondary Task: Single-Hop Fact QA
Answer questions based on local paragraphs.
Context Requirement: Partial | Metric: Accuracy

Output format:
Output the "[Answer]" identifier first, then output the answer option
letter (A/B/C/D), without any additional content.
[Answer] A
```

E.3.3 T2-c: Long-Range Entity and Commitment Tracking

T2-c: Long-Range Entity and Commitment Tracking

```
Primary Task: Dialogue Memory and Long-Horizon Tracking
Track and respond to dialogue history.

Secondary Task: Long-Range Entity and Commitment Tracking
Track entity states across the global context.
Context Requirement: Full | Metric: Accuracy

Output format:
Output the "[Answer]" identifier first, then output A/B/C/D/E.
[Answer] B
```

E.3.4 T2-d: Short-Range Reference Resolution

T2-d: Short-Range Reference Resolution

```
Primary Task: Dialogue Memory and Long-Horizon Tracking
Track and respond to dialogue history.

Secondary Task: Short-Range Reference Resolution
Resolve references and states in local context.
Context Requirement: Partial | Metric: Accuracy

Output format:
Output the "[Answer]" identifier first, then output the answer option
letter (A/B/C/D), without any additional content.
[Answer] B
```

E.4 T3: Exhaustive Retrieval

E.4.1 T3-a: Full-Sentence Citation Alignment

T3-a: Full-Sentence Citation Alignment

```
Primary Task: Attribution and Citation Alignment
Bind correct sources to generated text.

Secondary Task: Full-Sentence Citation Alignment
Context Requirement: Full | Metric: F1

Instruction: Identify the original Part number(s) that fully support
the given summary sentence.

Output format:
Output "[Answer]" first, then output cited parts line by line.
[Answer]
Part 1
Part 12
```

E.4.2 T3-b: Key-Statement Citation Alignment

T3-b: Key-Statement Citation Alignment

```
Primary Task: Attribution and Citation Alignment
Bind correct sources to generated text.

Secondary Task: Key-Statement Citation Alignment
Context Requirement: Partial | Metric: F1

Instruction: Identify the source location for the specified sentence
using paragraph identifiers from the source text.

Output format:
Output "[Answer]" first, then output paragraph identifiers line by line.
[Answer]
S1
S2
```

E.4.3 T3-c: Targeted Subset Identification

T3-c: Targeted Subset Identification

```
Primary Task: Aggregation and Clustering

Secondary Task: Targeted Subset Cluster Identification
Return instances matching a query category.
Context Requirement: Partial | Metric: F1

Output format:
Output "[Answer]" first, then output matching IDs line by line.
[Answer]
A22V1MD93T2FW9
ACJT8MUCOLRFO
```

E.4.4 T3-d: Global Conflict and Inconsistency Localization

T3-d: Global Conflict and Inconsistency Localization

```
Primary Task: Consistency and Compliance Checking

Secondary Task: Global Conflict and Inconsistency Localization
Context Requirement: Full | Metric: F1

Output format:
Output "[Answer]" first, then output inconsistent pairs as "A[id] B[id]".
[Answer]
A1 B2
```

E.4.5 T3-E: TARGETED RULE VIOLATION DETECTION

T3-e: Targeted Rule Violation Detection

```
Primary Task: Consistency and Compliance Checking

Secondary Task: Targeted Rule or Condition Violation Detection
Context Requirement: Partial | Metric: F1

Output format:
Output "[Answer]" first, then output violated term IDs line by line.
[Answer]
5
6
```

E.4.6 T3-F: COMPREHENSIVE ERROR AND ANOMALY SWEEP

T3-f: Comprehensive Error and Anomaly Sweep

```
Primary Task: Consistency and Compliance Checking

Secondary Task: Comprehensive Error and Anomaly Sweep
Context Requirement: Full | Metric: F1

Output format:
Output "[Answer]" first, then output all identified errors line by line.
[Answer]
beleive
freind
```

E.4.7 T3-G: MULTI-VERSION IMPACT ANALYSIS

T3-g: Multi-Version Impact Analysis

```
Primary Task: Version and Code Diff Analysis

Secondary Task: Dependency-Aware Multi-Version Impact Analysis
Context Requirement: Full | Metric: F1

Instruction: Identify all methods/elements that changed status between
versions. Use the format "MethodName(ParameterTypes)".

Output format:
Output "[Answer]" first, then output identifiers line by line.
[Answer]
MethodA(java.lang.String)
MethodB()
```

E.4.8 T3-H: LOCALIZED INTERFACE CHANGE DETECTION

T3-h: Localized Interface Change Detection

```
Primary Task: Version and Code Diff Analysis

Secondary Task: Localized Interface Change Detection
Context Requirement: Partial | Metric: F1

Instruction: Identify fields renamed, removed, or refactored between
versions. Use the format "Model: [Field Name]".

Output format:
Output "[Answer]" first, then output fields line by line.
[Answer]
Lesson Model: key
```

E.5 T4: NUMERICAL REASONING

E.5.1 T4-A: MULTI-SOURCE CONSISTENCY VERIFICATION

T4-a: Multi-Source Consistency Verification

```
Primary Task: Structured and Numeric Reasoning

Secondary Task: Structured Multi-Source Consistency Verification
Context Requirement: Full | Metric: math_verify

Instruction: Given multiple tables, use the provided formula to determine
whether computed values are consistent. Return "No Error" or the
corresponding record IDs.

Output format:
Output "[Answer]" first.
[Answer]
No Error
```

E.5.2 T4-B: SINGLE-SOURCE TARGETED AGGREGATION

T4-b: Single-Source Targeted Aggregation

```
Primary Task: Structured and Numeric Reasoning

Secondary Task: Single-Source Targeted Aggregation
Context Requirement: Partial | Metric: math_verify

Instruction: Calculate the requested quantities from the specified table.

Output format:
Output "[Answer]" first, then computed values line by line.
[Answer]
5.5%
-2.2%
```

E.5.3 T4-C: PROCEDURAL STATE TRACKING

T4-c: Procedural State Tracking

```
Primary Task: Structured and Numeric Reasoning

Secondary Task: Long-Context Procedural State Tracking
Context Requirement: Full | Metric: math_verify

Output format:
Output "[Answer]" first, then tracked states line by line.
[Answer]
state_1
state_2
```

E.6 T5: STRUCTURED EXTRACTION

T5: Multi-Table Structured Extraction

```
Primary Task: Structured Table Reasoning

Secondary Task: Long-Context Multi-Table Integration
Context Requirement: Full | Metric: IoU-based Structured Match

Instruction: Given multiple tables and a question, identify join keys,
reason step by step, and produce the answer as a valid JSON object.

Output format:
Output the final answer between <answer> and </answer> tags.
{"columns": ["col1", ...], "data": [["val1", ...], ...]}
```

E.7 T6: Structured Matching

E.7.1 T6-a: Large-Scale Document Clustering

T6-a: Large-Scale Document Clustering

```
Primary Task: Aggregation and Clustering

Secondary Task: Large-Scale Document Clustering
Context Requirement: Full | Metric: SubEM

Instruction: Cluster the given documents by the specified criterion.

Output format:
Output "[Answer]" first, then "ClusterName Proportion%" line by line.
[Answer]
Cluster_A 25.00%
Cluster_B 75.00%
```

E.7.2 T6-b: Large-Scale In-Context Rule Induction

T6-b: In-Context Rule Induction

```
Primary Task: Rule Induction and In-Context Learning

Secondary Task: Large-Scale In-Context Rule Induction
Context Requirement: Full | Metric: SubEM

Instruction: Apply the induced rules from the provided text to reformat
or transform the given input snippet.

Output format:
Output "[Answer]" first, then the fully formatted result.
[Answer]
formatted output
```

E.7.3 T6-c: Targeted Example-Based Rule Induction

T6-c: Targeted Rule Induction

```
Primary Task: Rule Induction and In-Context Learning

Secondary Task: Targeted Example-Based Rule Induction
Context Requirement: Partial | Metric: SubEM

Instruction: Based on case examples in the specified section, answer the
given question by applying the induced rule.

Output format:
Output "[Answer]" first, then the answer.
[Answer]
support
```

E.8 T7: Relevance Ranking

E.8.1 T7-a: Global Cohesive Retrieval

T7-a: Global Cohesive Retrieval

```
Primary Task: Retrieval and Ranking

Secondary Task: Global Cohesive Retrieval
Context Requirement: Full | Metric: NDCG@k

Instruction: Retrieve all items meeting the specified condition and
output their IDs in the required sorted order.

Output format:
Output "[Answer]" first, then IDs line by line.
```

```
[Answer]
A2AV7Q95QGPTO0
A3NM0RAYSL6PA8
```

E.8.2 T7-b: Key-Snippet Retrieval

T7-b: Key-Snippet Retrieval

```
Primary Task: Retrieval and Ranking

Secondary Task: Key-Snippet Retrieval
Context Requirement: Partial | Metric: NDCG@k

Instruction: From the specified subset, retrieve items matching the
given category and sort by the requested criterion.

Output format:
Output "[Answer]" first, then sorted IDs line by line.
[Answer]
ID1
ID2
```

E.9 T8: Sequence Ordering

E.9.1 T8-a: Global Timeline Reconstruction

T8-a: Global Timeline Reconstruction

```
Primary Task: Sequencing and Structure Reconstruction

Secondary Task: Global Timeline Reconstruction
Context Requirement: Full | Metric: Pairwise Accuracy

Instruction: The text is divided into multiple parts in shuffled order.
Sort these parts into the correct chronological sequence.

Output format:
Output "[Answer]" first, then part identifiers in order.
[Answer]
Part 5
Part 1
Part 9
```

E.9.2 T8-b: Local Causal Chain Sorting

T8-b: Local Causal Chain Sorting

```
Primary Task: Sequencing and Structure Reconstruction

Secondary Task: Local Causal Chain Sorting
Context Requirement: Partial | Metric: Pairwise Accuracy

Instruction: The specified paragraph is out of order. Reorder it
according to causal or logical structure.

Output format:
Output "[Answer]" first, then sorted option letters line by line.
[Answer]
A
B
C
```

E.9.3 T8-C: GLOBAL FREQUENCY ANALYSIS

T8-c: Global Frequency Analysis

```
Primary Task: Aggregation and Clustering

Secondary Task: Global Frequency Analysis
Context Requirement: Full | Metric: Pairwise Accuracy

Instruction: Sort the given terms in descending order by their frequency
of appearance in the text.

Output format:
Output "[Answer]" first, then sorted terms line by line.
[Answer]
would
this
that
```

E.10 T9: SUMMARIZATION

E.10.1 T9-A: GLOBAL-COVERAGE CONSTRAINED SUMMARY

T9-a: Global-Coverage Constrained Summary

```
Primary Task: Summarization and Synthesis

Secondary Task: Global-Coverage Constrained Summary
Context Requirement: Full | Metric: ROUGE-L

Output format:
Output "[Answer]" first, then the summary.
[Answer]
your summary
```

E.10.2 T9-B: QUERY-FOCUSED SUMMARY

T9-b: Query-Focused Summary

```
Primary Task: Summarization and Synthesis

Secondary Task: Query-Focused Summary
Context Requirement: Partial | Metric: ROUGE-L

Output format:
Output "[Answer]" first, then the summary.
[Answer]
your summary
```