Incentivizing In-depth Reasoning over Long Contexts with Process Advantage Shaping

Miao Peng[1*], **Weizhou Shen**[2], **Nuo Chen**[1], **Chenliang Li**[2], **Ming Yan**[2†], **Jia Li**[1†]
[1]The Hong Kong University of Science and Technology (Guangzhou)
[2]Tongyi Lab, Alibaba Group
{mpeng885,jialee}@connect.hkust-gz.edu.cn
ym119608@alibaba-inc.com

Abstract

Reinforcement Learning with Verifiable Rewards (RLVR) has proven effective in enhancing LLMs' short-context reasoning but falters in long-context scenarios requiring precise grounding and multi-hop reasoning. We identify the "almost-there" phenomenon—trajectories that are largely correct but fail at the final step—in long-context reasoning RL and attribute this failure to two factors: (1) the lack of high reasoning density in long-context QA data, and (2) indiscriminate penalization of partially correct trajectories during long-context RL. To overcome this bottleneck, we propose **DEEPREASONQA**, a KG-driven synthesis framework that controllably constructs high-difficulty, multi-hop long-context QA pairs with inherent reasoning chains. Building on this, we introduce Long-context Process Advantage Shaping (**LONGPAS**), a simple yet effective method that performs fine-grained credit assignment by measuring reasoning steps along *Validity* and *Relevance* dimensions, which captures critical signals from "almost-there" trajectories. Experiments on three long-context reasoning benchmarks show that our approach substantially outperforms RLVR baselines and matches frontier LLMs while using far fewer parameters. Further analysis confirms the effectiveness of our methods in strengthening long-context reasoning while maintaining stable RL training. Codes are available at: `https://github.com/GKNL/LongPAS`.

1 Introduction

Reasoning over long contexts is a critical capability for modern large language models (LLMs), as many real-world tasks—such as document understanding (Bai et al., 2024b, 2025) or agentic deep research (Jin et al., 2025; Team et al., 2025)—require grounding information and perform complex reasoning across millions of tokens (Hsieh et al.; Ling et al., 2025; Krishna et al., 2025). While advanced LLMs have successfully employed RLVR (Reinforcement Learning with Verifiable Rewards) to enhance short-context complex reasoning (Guo et al., 2025; Yue et al., 2025; Peng et al., 2025), performance still degrades significantly when confronted with long-context scenarios. Recent works have explored improving long-context capability in RL by progressively scaling the context window (Wan et al., 2025), or employing data-driven RL to learn advanced reasoning patterns (Wang et al., 2025; Chen et al., b). However, these methods primarily focus on improving information grounding but still struggle with in-depth reasoning over long-context documents.

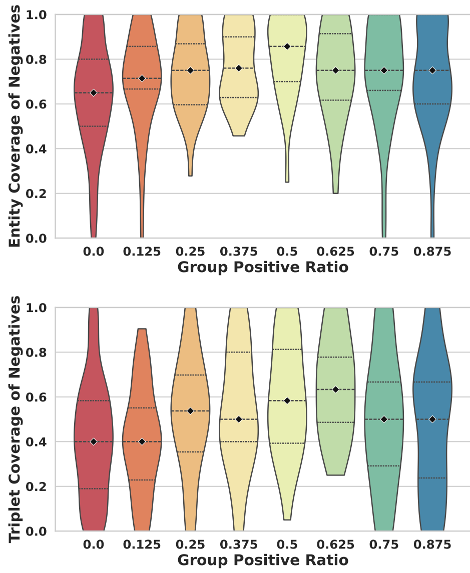


Figure 1: Entity & Triplet Coverage between negative rollouts and GT reasoning chains on FRAMES (Krishna et al., 2025) with Qwen3-4B model.

A key reason is that they suffer from the limitation of current outcome-based RLVR algorithms (Shao et al., 2024; Sheng et al., 2024a; Cheng et al., 2025), which uniformly rely on sparse, outcome-level rewards. This makes it

*Research during internship at Tongyi Lab, Alibaba Group
†Corresponding author

Findings of the Association for Computational Linguistics: ACL 2026, pages 26201–26228
July 2-7, 2026

hard to distinguish the common "almost-there" cases—trajectories where most reasoning steps are correct but the final answer is wrong—causing RLVR to discard valuable learning signals, especially in complex long-context tasks where such cases are frequent. Empirically, our analysis of Entity & Triple Coverage across uncertainty levels in Figure 1 shows that LLM correctly anchors most critical entities but fails at integrating them for final reasoning (Further analysis in Section 4.1). Despite their potential to provide valuable learning signals (Bae et al., 2025), these trajectories are penalized as failures under sparse reward regimes.

While process-level supervision is a natural remedy for reward sparsity, long-context tasks introduce unique challenges: (1) **Sparsity Reasoning Density**: Unlike logic-dense short-context tasks, critical information in long documents is highly scattered. This "grounding-reasoning imbalance" makes it difficult for LLMs to incentivize high-quality reasoning patterns amidst contextual noise. Besides, the vast search space of long-form text hinders effective construction of high-quality, step-level data. (2) **Indeterminate Credit Assignment**: Since many steps involve mere context grounding, it is impractical to isolate which specific step contributed to an "almost-there" failure, complicating fine-grained supervision assignment. These challenges motivate the following research questions:

1. How to construct long-context QA data with high reasoning density, explicit step dependencies, and reliable step-level supervision?

2. How can step-level signals be integrated into long-context RL to achieve fine-grained credit assignment?

To resolve these problems, we shift the paradigm from mining supervision in noisy natural data to constructing supervision via controlled synthesis. We first introduce a KG-driven synthesis framework to automatically generate **DEEPREASONQA**, a large-scale, high-difficulty multi-hop QA dataset from sparse long documents at scale. This controllable process inherently provides explicit reasoning chains, offering reliable step-level supervision. Building on this, we propose Long-context Process Advantage Shaping (**LONGPAS**) to improve credit assignment in long-context RL. By leveraging reference chains, LONGPAS measures reasoning steps across *Validity* and *Relevance* dimensions to compute token-level advantage reweighting coefficients. This fine-grained shaping prevents the penalization of "almost-there" trajectories, stabilizing the learning of complex reasoning. Comprehensive evaluations across in-domain and out-of-domain benchmarks demonstrate that LONGPAS significantly outperforms RLVR baselines across multiple LLM series and rivals frontier LLMs with far more parameters. Beyond accuracy, LONGPAS induces more effective reasoning policies with precise grounding and robust long-range logic, ensuring stable RL training and stepwise reasoning validity in challenging long-context scenarios.

2 Related Works

Long-Context Data Synthesis Early long-context synthesis primarily extended input lengths by augmenting short-context datasets with distracting or irrelevant content (Li et al., 2024a,b,c; Zhang et al.; Chen et al., 2024). While effective for increasing context window, they fail to provide the high-density reasoning pattern necessary for in-depth reasoning. Recent advancements focus on generating more complex long contexts. For instance, LongFaith (Yang et al., 2025b) utilizes citation-based prompting for faithfulness, while Wildlong (Li et al., 2025) employs graph-based modeling for realism. Others generate extended contexts for existing pairs (Zhu et al., 2025a), use query-centric document aggregation (Gao et al., a), or transform short documents into coherent long-form data via semantic retrieval and reordering (Zhang et al., 2025; Gao et al., b).

Long-Context Reasoning Existing models have extended context windows via techniques like RoPE (Su et al., 2024; Peng et al., 2023) during pre-training (Yang et al., 2025a; Deepmind, 2025), but they often struggle with in-depth reasoning over long contexts. Recent efforts have explored post-training to unlock long-context potential. Methods such as Long-context SFT (Bai et al., 2024a) and DPO (Chen et al., a) frequently introduce non-generalizable biases. More recent approaches like QwenLong-L1 & L1.5 (Wan et al., 2025; Shen et al., 2025), SoLoPO (Sun et al., 2025), and E-GRPO (Zhao et al., 2025) mainly focus on RL strategy optimization while overlooking the necessity of high-quality synthetic reasoning data. LoongRL (Wang et al., 2025) addresses the data gap by adding distracting documents to multi-hop questions. However, they all remain susceptible to the phenomenon where outcome-based rewards discard valuable signals in long-context scenarios.

3 Preliminary

Given a question Q and a long context C, standard long-context RLVR framework optimizes a policy $\pi_\theta(y \mid C, Q)$ to maximize the expected verifiable reward $r_{\text{ans}}(y)$ of the final answer: $J(\theta) = \mathbb{E}_{y\sim\pi_\theta}[r(y)]$. We employ Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which eliminates the need for a critic model by estimating advantages within sampled groups. Given a prompt $(C, Q) \sim \mathcal{D}$, GRPO first samples a group of G candidate answers $\boldsymbol{y} = \{y_1, \ldots, y_G\}$ from the old policy $\pi_{old}(y \mid C, Q)$. For each y_i, the advantage $\hat{A}_i$ is computed by normalizing its reward r_i against the group mean and standard deviation: $\hat{A}_i = (r_i - \text{mean}(\boldsymbol{r}))/\text{std}(\boldsymbol{r})$. The policy π_θ is optimized by maximizing the following objective:

$$J_{\text{GRPO}}(\theta) = \hat{\mathbb{E}}_{(C,Q),\boldsymbol{y}}\left[\frac{1}{G}\sum_{i=1}^{G} f_\epsilon\left(\rho_i(\theta), \hat{A}_i\right)\right] - \beta \cdot \hat{\mathbb{E}}_{(C,Q)}\left[\mathbb{D}_{KL}[\pi_\theta(\cdot \mid C,Q) \| \pi_{old}(\cdot \mid C,Q)]\right],$$

where $\rho_i(\theta) = \frac{\pi_\theta(y_i|C,Q)}{\pi_{old}(y_i|C,Q)}$ is the importance weight. $f_\epsilon(x, A) = \min(xA, \text{clip}(x, 1-\epsilon, 1+\epsilon)A)$ is the PPO clipping function, and β is a hyperparameter controlling the KL divergence penalty.

4 Methodology

4.1 The Challenge of Almost-there Phenomenon in Long-Context Reasoning

Sparse rewards often hinder RLVR on complex long-context tasks. Since long trajectories comprise multiple components (e.g., grounding, reasoning), final-answer-based penalties may indiscriminately suppress valid sub-steps within incorrect rollouts, degrading overall performance. To quantify the prevalence of correct reasoning steps within the long-context multi-hop trajectories, we conduct an empirical analysis on FRAMES (Krishna et al., 2025). We first use Gemini-2.5-Pro to identify ground-truth reasoning chains P for each QA pair (Q, A), represented as a sequence of triplets (s_i, r_i, o_i). For each question, we perform N rollouts $T_1, T_2, ..., T_N$ and group them by the ratio of positive outcomes. Subsequently, for each negative rollout T_f, we calculate the coverage metrics using the following formulas:

Entity Coverage measures the proportion of correct entities in P that are present in T_f. It is calculated as $\frac{|\mathcal{E}(T_f)\cap\mathcal{E}(P)|}{|\mathcal{E}(P)|}$, where $\mathcal{E}(T_f)$ and $\mathcal{E}(P)$ are the sets of entities in T_f and P, respectively.

Triplet Coverage uses an LLM-as-a-judge to evaluate the correctness of each step in T_f. We define $\mathcal{R}(T_f)$ as the set of triplets in T_f, and $\mathbb{I}_{Judge}(R, P)$ as an indicator that judges if the triplet $R \in \mathcal{R}(T_f)$ is a valid and correct reasoning step present in P. The Triplet Coverage is then calculated as: $\frac{\sum_{R\in\mathcal{R}(T_f)} \mathbb{I}_{Judge}(R,P)}{|\mathcal{R}(P)|}$.

Figure 1 illustrates the coverage distributions and the results show that: as the Group Positive Ratio increases (i.e., questions become easier), both coverage metrics for negative rollouts rise, indicating that many failed trajectories contain substantial correct reasoning. The consistently higher Entity than Triplet Coverage further suggests that, while the model excels at grounding but struggle with logical chaining. Notably, coverage peaks around 50% success ratio, representing borderline cases that are especially valuable for RL (Bae et al., 2025). This "almost-there" phenomenon necessitates reward mechanisms that recognize and preserve valid sub-steps even in failed outcomes.

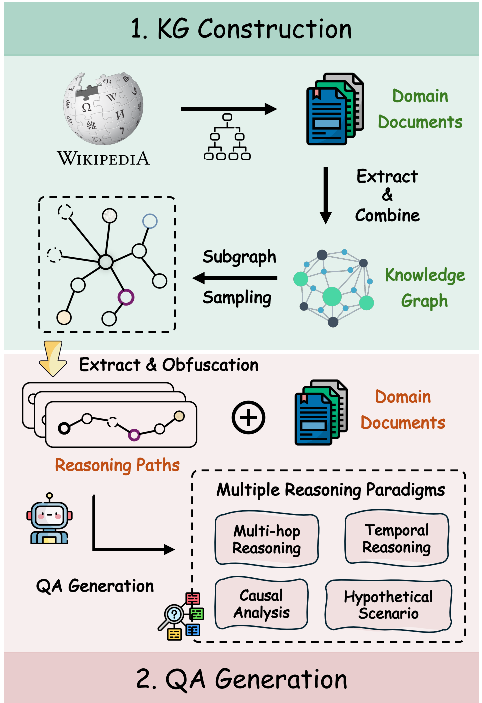


Figure 2: Overall pipeline of the knowledge-guided long-context multi-hop QA synthesis framework.

4.2 Multi-hop Reasoning QA Synthesis

To address the scarcity of high-reasoning-density QA pairs that necessitate long-range reasoning over documents, we propose a Knowledge-Guided

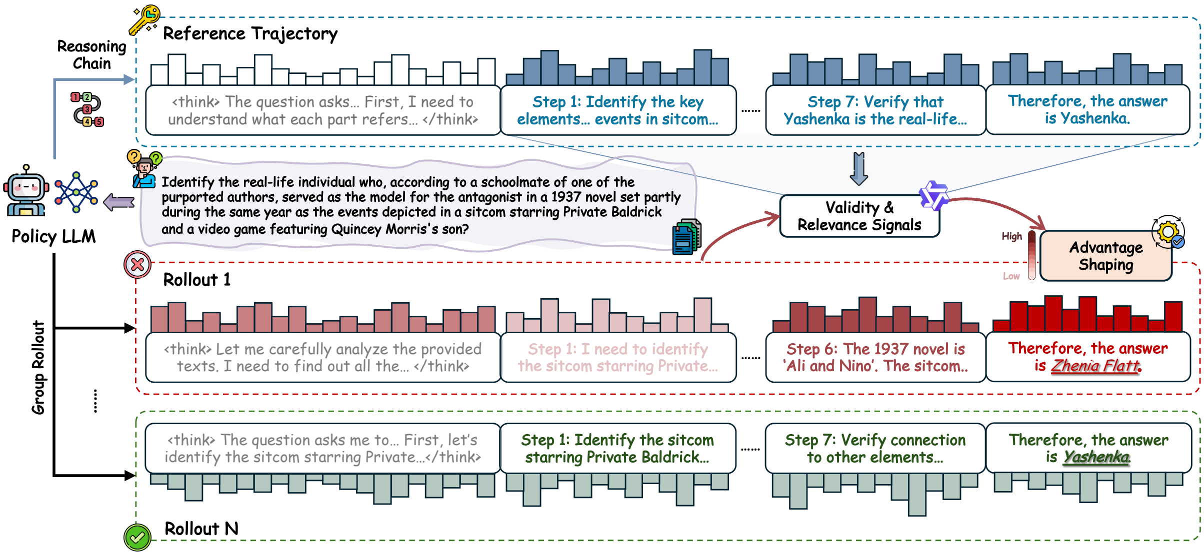


Figure 3: Overview of Long-Context Reinforcement Learning with Process Advantage Shaping.

Long-Context Multi-hop QA Synthesis Framework. As shown in Figure 2, it automatically extracts and constructs complex multi-hop QA pairs from noisy long documents and simultaneously produces high-quality reasoning chains that provide explicit dependency paths for stepwise supervision.

KG Construction We first collect Wikipedia page documents according to the hierarchical catalog, spanning diverse knowledge domains. From each document, we extract triplets and merge them into an initial knowledge graph G (Mo et al., 2025; Jiang et al., 2025). A more complex, cross-document knowledge graph G_d is subsequently formed via domain-level aggregation. To enhance the quality and coherence of the graph, G_d is further refined through entity and relation clustering.

Reasoning Path Sampling To generate challenging multi-hop paths, we first sample relation-relevant subgraphs centered on target entities within the domain-specific KG. We then derive long-range reasoning paths using strategies such as Random Walk and BFS, under the constraint that key nodes are sparsely distributed across many documents to enforce cross-document grounding and retrieval. These sampled paths are further hardened through information perturbation and entity/concept obfuscation in G_d, e.g., Temporal Obfuscation ("the year ending with 5 in the late 20th century") and Location Obfuscation ("a country with a population over 1.4 billion").

Question Generation We identify that high-reasoning-density questions necessitate the ability to synthesize deep and widely scattered contextual information for long-range reasoning. To this end, we categorize deep reasoning in long-context scenarios into four distinct paradigms: **Multi-hop Reasoning**, **Causal Analysis**, **Temporal Reasoning** and **Hypothetical Scenario**. Using source documents and extracted reasoning paths as ground truth, we employ a strong teacher LLM to synthesize high-quality multi-hop QA pairs that integrate these paradigms and apply question obfuscation. Question complexity is controlled by the length of sampled paths, and dataset diversity is ensured by using paths from multiple sampling strategies.

Quality Control To assure the synthesized QAs are paired with well-grounded documents, concise answers and high-quality reasoning paths, we apply a four-stage quality control pipeline: (1) *Answer Alignment Check*: teacher LLMs act as generator, responder, and verifier to ensure question–answer consistency; (2) *Knowledge Grounding Check*: we discard questions answerable without the source documents to mitigate reliance on parametric knowledge; (3) *Complex Answer Filtering*: we keep only QA pairs with answers under 20 words to ensure reliable verification; (4) *Contextual Robustness Check*: samples that are easily perturbed by irrelevant documents are filtered out. The detailed pipeline is provided in Appendix A.2.

4.3 Long-Context Reinforcement Learning with Process Advantage Shaping

To effectively address the common "almost-there" phenomenon in long-context RL, we introduce

LONGPAS, which incorporates process advantage shaping guided by on-policy reference trajectories to facilitate fine-grained credit assignment. The overall framework is illustrated in Figure 3.

On-policy GT-guided Rollout Given a question Q and long context C, in addition to regular group sampling rollouts T, we perform auxiliary sampling by prompting the LLM to transform ground-truth (GT) entity-relation triplets into a natural language trajectory τ_p. This ensures τ_p reflects the model's own reasoning patterns, serving as an on-policy reference. To facilitate parsing, we use a step-by-step prompt template (see Appendix B) to enforce a structured, intermediate-step format.

Hybrid Reward Design For open-ended long-context QA, rule-based verification alone often fails due to answer diversity. We adopt a hybrid reward mechanism (Wan et al., 2025) to balance precision and recall. Specifically, we compute the final reward as the maximum of a rule-based exact match and an LLM-based semantic judge:

$$R(Q, y_{pred}, y_{gt}) = \max\{\mathbb{I}(y_{pred} = y_{gt}), \\ \text{LLM}_{Judge}(Q, y_{pred}, y_{gt})\}.$$

where $\mathbb{I}(\cdot)$ is the indicator function for string matching. This hybrid approach mitigates false negatives by allowing the LLM judge to recognize semantically equivalent but syntactically different answers.

Process Advantage Shaping We posit that the reference trajectory τ_p encapsulates the necessary grounded information and reasoning logic. To mitigate erroneous penalization of valid sub-steps, we perform stepwise advantage shaping for negative rollouts T_{neg}, while leaving positive rollouts intact to encourage exploration. Specifically, for group sampling rollouts $T = \{\tau_1, \tau_2, ..., \tau_n\}$, we evaluate each sub-step s_j of rollout τ_i in two dimensions: *validity* and *relevance*. *Validity* is evaluated with an LLM-as-a-Judge, denoted as $\mathbb{I}_{\text{valid}}(s_{i,j}) = \text{LLM}_{Judge}(s_j, \tau_p)$. The judge assesses whether τ_i aligns with the necessary entities and reasoning logic of the reference τ_p. *Relevance* is quantified by semantic similarity $\text{sim}(s_{i,j}, \tau_p)$ and reflects the extent to which the sub-step s_j is semantically aligned with its rollout τ_i. The reweighted stepwise advantage $\hat{A}_{i,j}$ for the j-th step $s_{i,j}$ in the i-th rollout τ_i is calculated as:

$$\hat{A}_{i,j} = \underbrace{\frac{r_i - \text{mean}(\{r_1, r_2, ..., r_n\})}{\text{std}(\{r_1, r_2, ..., r_n\})}}_{\text{Group Relative Advantage}} \\ \cdot \underbrace{(1 - \mathbb{I}(\tau_i \in T_f) \cdot \mathbb{I}_{\text{valid}}(s_{i,j}) \cdot \text{sim}(s_{i,j}, \tau_p))}_{\text{Step-wise Reweighting Coefficient}},$$

where $\mathbb{I}(\tau_i \in T_f)$ is an indicator for negative rollouts, and $\text{sim}(s_{i,j}, \tau_p)$ denotes the semantic similarity between step $s_{i,j}$ and τ_p. Consequently, when a sub-step in a negative rollout is deemed correct, the penalty vanishes ($\hat{A}_{i,j} \to 0$), whereas erroneous steps retain the full negative signal.

We opt to mitigate penalties rather than assign positive rewards to avoid optimization ambiguity. Positively reinforcing sub-steps in failed rollouts risks incentivizing "plausible but ineffective" paths. Instead, our conservative credit assignment signals that correct sub-steps were not the cause of failure, preventing valid reasoning from being "unlearned" without falsely labeling it as sufficient for success.

Overall Training Objective Following the defined Process Advantage Estimation algorithm, we now formalize the final training objective, which incorporates the Step-wise Re-weighted Advantage ($\hat{A}_{i,j}$). The policy π_θ is optimized by maximizing the following objective function:

$$J_{\text{GRPO}}(\theta) = \hat{\mathbb{E}}_{(C,Q),\boldsymbol{y}} \left[\frac{1}{N} \sum_{i=1}^{N} \frac{1}{L_i} \sum_{j=1}^{L_i} f_\epsilon \left(\rho_{i,j}(\theta), \hat{A}_{i,j} \right) \right] \\ - \beta \cdot \hat{\mathbb{E}}_{(C,Q)} \left[\mathbb{D}_{KL}[\pi_\theta(\cdot \mid C, Q) \| \pi_{old}(\cdot \mid C, Q)] \right],$$

where $\rho_{i,j}(\theta) = \frac{\pi_\theta(s_{i,j}|s_{i,<j},C,Q)}{\pi_{old}(s_{i,j}|s_{i,<j},C,Q)}$ denotes the probability ratio of j-th step in i-th trajectory.

5 Experimental Setup

Training Settings We construct the DEEPREASONQA training dataset containing 2,012 QA samples with documents up to 60K tokens. Dataset construction details and statistics are provided in Appendix A.1 and A.2. We conduct experiments on different LLM backbones: (1) Instruct Models: LLaMA3.1-8B-Instruct, Qwen2.5-7B-Instruct, Qwen3-4B-Instruct and Qwen3-30B-A3B-Instruct; (2) Thinking Models: Qwen3-4B-Thinking and Qwen3-30B-A3B-Thinking. We build our RL framework on VeRL (Sheng et al., 2024b). Training uses AdamW optimizer with learning rate 2e-6 and a 5-step linear warmup. The max input length is

Models	FRAMES	LongBench V2					Multi-Hop QA			
		SingleDoc	MultiDoc	Code Repo	Dialogue	**Overall**	2Wiki	HotpotQA	MusiQue	**Avg**
Frontier Models										
GPT5-Nano	73.54	44.00	39.20	50.00	46.15	43.74	89.00	82.00	61.50	77.50
Gemini-2.5-Flash-Thinking	65.78	51.43	55.20	58.00	66.67	56.77	89.34	81.00	60.00	76.78
GPT-OSS-120B	72.69	44.57	43.20	53.06	61.54	47.01	89.00	82.00	66.00	79.00
GPT-OSS-20B	64.44	38.51	40.80	56.00	61.54	43.37	88.50	79.00	49.25	72.25
Instruct Models										
LLaMA3.1-8B-Instruct	41.84	28.86	26.20	27.00	32.05	27.93	43.62	61.75	21.00	42.12
- RLVR	55.98	31.86	29.20	29.00	32.69	28.93	68.00	69.75	46.62	61.46
- LongPAS	**60.62**	30.14	29.80	32.00	34.62	**29.62**	79.38	76.38	52.00	**69.25**
Qwen2.5-7B-Instruct	45.08	36.71	28.20	32.00	35.90	33.60	54.75	69.75	32.38	52.29
- RLVR	50.97	34.43	29.00	30.00	40.38	31.01	68.25	67.75	40.25	58.75
- LongPAS	**55.76**	39.57	28.40	29.50	31.41	**33.70**	80.88	80.00	51.62	**70.83**
Qwen3-4B-Instruct	46.81	36.00	30.60	36.50	60.26	37.28	63.25	67.62	28.75	53.21
- RLVR	60.92	39.14	37.00	49.50	60.90	42.10	82.62	75.88	49.25	69.25
- LongPAS	**64.90**	38.14	40.60	48.00	63.46	**42.94**	86.50	75.38	55.00	**72.29**
Qwen3-30B-A3B-Instruct	62.96	42.43	38.40	52.00	62.82	44.43	83.88	77.62	51.50	71.00
- RLVR	66.26	50.29	50.40	42.00	56.41	47.91	86.00	76.00	57.00	**73.00**
- LongPAS	**68.93**	46.29	42.40	56.00	71.79	**49.11**	84.50	77.50	55.50	72.50
Reasoning Models										
Qwen3-4B-Thinking	60.84	37.00	35.60	41.50	60.26	40.46	85.50	75.12	50.25	70.29
- RLVR	62.74	37.14	40.80	46.00	66.67	41.75	83.50	71.50	53.50	69.50
- LongPAS	**64.75**	40.29	40.00	43.00	62.82	**42.30**	85.62	78.62	54.25	**72.83**
Qwen3-30B-A3B-Thinking	69.66	44.00	44.00	46.00	64.10	48.31	84.31	76.94	63.44	74.90
- RLVR	69.66	40.00	46.40	50.00	51.28	44.53	84.00	77.00	64.00	75.00
- LongPAS	**71.84**	47.43	54.40	60.00	76.92	**54.27**	86.50	77.00	66.00	**76.50**

Table 1: Overall performance of models on long-context QA benchmarks. RLVR is implemented with GRPO (Shao et al., 2024). The top scores for each backbone LLM are **bolded**. We additionally report four representative reasoning-intensive sub-task performance of LongBench V2, full results are provided in Appendix C.7.

60K tokens; max output length is 30K for Thinking models and 10K for Instruct models. We conduct purely on-policy training with batch size 128 and set group size N to 8, sampling temperature to 0.7 and top-p to 0.95. During GT path guided sampling, temperature is set to 0 and top-p is set to 1. More implementation details are listed in Appendix B.

Evaluation Configurations We evaluate all LLMs on three widely-used and challenging long-context QA benchmarks: (1) FRAMES (Krishna et al., 2025), which comprises questions requiring 2-15 Wikipedia articles to answer. (2) LongBench V2 (Bai et al., 2025), a realistic multi-choice QA benchmark containing long-context problems requiring deep understanding and reasoning, with contexts ranging from 8K to 2M words. (3) Multi-Hop QA: We adopt three subsets 2WikiMultiHopQA, HotpotQA and MusiQue from Longbench (Bai et al., 2024b), which cover 3-5 hop questions with corresponding documents. To contextualize performance, we also evaluate frontier LLMs in different sizes including GPT5-Nano (OpenAI, 2025), Gemini2.5-Flash-Thinking (Deepmind, 2025) and GPT-OSS (20B and 120B) (Agarwal et al., 2025). More detailed evaluation configurations can be found in Appendix B.

6 Experimental Results

6.1 Main Results

In the section, we compare LongPAS with different LLM series and training strategies. More results on **length generalization** can be found in Appendix C.1.

Substantial Gains over Various LLMs Table 1 reports the main results of LongPAS compared to various baselines and frontier LLMs across different model families and scales (4B, 8B, 30B, etc.). We highlight two key observations: (1) Both our synthesized dataset DeepReasonQA and our training method LongPAS contribute substantial improvements. For example, RLVR yields significant gains over the original LLMs across all benchmarks: Qwen3-4B-Instruct (46.81 vs. 60.92 on FRAMES), Qwen3-4B-Thinking (60.84 vs. 62.74 on FRAMES). These improve-

Models	FRAMES	LongBench V2					Multi-Hop QA			
		SingleDoc	MultiDoc	Code Repo	Dialogue	**Overall**	2Wiki	HotpotQA	MusiQue	**Avg**
Qwen3-4B-Instruct	46.81	36.00	30.60	36.50	60.26	37.28	63.25	67.62	28.75	53.21
- SFT	55.10	28.57	30.40	26.00	41.03	31.41	79.50	74.00	49.00	67.50
- GRPO	60.92	39.14	37.00	49.50	60.90	42.10	82.62	75.88	49.25	69.25
- DAPO	62.11	37.57	36.60	43.00	58.33	40.26	85.00	76.00	51.12	70.71
- **LONGPAS (Ours)**	**64.90**	38.14	40.60	48.00	63.46	**42.94**	86.50	75.38	55.00	**72.29**
Qwen3-4B-Thinking	60.84	37.00	35.60	41.50	60.26	40.46	85.50	75.12	50.25	70.29
- SFT	59.95	37.14	31.20	36.00	53.85	35.19	84.50	72.50	49.50	68.83
- GRPO	62.74	37.14	40.80	46.00	66.67	41.75	83.50	71.50	53.50	69.50
- DAPO	62.01	31.43	36.80	42.00	53.85	35.98	86.00	77.00	53.00	72.00
- **LONGPAS (Ours)**	**64.75**	40.29	40.00	43.00	62.82	**42.30**	85.62	78.62	54.25	**72.83**

Table 2: Pass@1 Performance Comparison of different training strategies on long-context QA benchmarks. For SFT training, we distill high-quality QAs with reasoning trajectories through DeepSeek-V3 and DeepSeek-R1 (for Instruct and Thinking model individually) under the guidance of GT reasoning chains.

Models	FRAMES	LongBench V2					Multi-Hop QA			
		SingleDoc	MultiDoc	Code Repo	Dialogue	**Overall**	2Wiki	HotpotQA	MusiQue	**Avg**
LONGPAS	**64.90**	38.14	40.60	48.00	63.46	**42.94**	86.50	75.38	55.00	**72.29**
- *w/o Validity* Signal	62.68	37.71	33.60	46.00	61.54	41.15	86.00	72.50	46.50	68.33
- *w/o Relevance* Signal	63.17	40.29	35.20	49.00	62.18	41.95	85.00	74.75	51.00	70.25
- *w/o* On-policy Supervision	62.89	40.57	38.40	47.50	65.38	42.45	85.50	76.62	52.88	71.67

Table 3: Ablation Study of LONGPAS on (a) **Process Signals** and (b) **On-policy Supervision**.

ments demonstrate the effectiveness of DEEPREASONQA. Furthermore, replacing vanilla RLVR with LONGPAS leads to additional gains: Qwen3-4B-Instruct (60.92 vs. 64.90 on FRAMES), Qwen3-4B-Thinking (69.50 vs. 72.83 on Multi-Hop QA); (2) LONGPAS achieves comparable performance to frontier LLMs with much fewer parameter scales. For example, LONGPAS trained on Qwen3-4B (both Instruct and Thinking) attains results on FRAMES and Multi-Hop QA that are comparable with Gemini-2.5-Flash-Thinking and GPT-OSS-20B. In addition, LONGPAS trained on Qwen3-30B-A3B surpasses GPT5-Nano and GPT-OSS-120B on LongBench V2.

Outperform different Training Strategies To further investigate how LONGPAS outperforms baseline training approaches, we report the comparison results against SFT, GRPO and DAPO in Table 2. It is evidenced that LONGPAS achieves larger gains than standard RL approaches like GRPO and DAPO (Sheng et al., 2024a) on both Instruct and Thinking models. Given that LONGPAS exploits ground-truth reasoning chains, we also include a SFT baseline: we distill reasoning trajectories from a teacher LLM (Gemini2.5-Pro) under the guidance of GT reasoning chains, and then supervised fine-tune the student LLM. However, SFT lags behind the RL-based methods, which exploit GT reasoning chains in an on-policy manner, highlighting the advantage of RL-style process optimization over offline distillation. Furthermore, the diminished performance on LongBench V2 suggests that SFT suffers from limited generalization to out-of-domain long-context tasks.

Consistent Improvements at Increasing Reasoning Depth To further investigate the effectiveness of our proposed LONGPAS in different long-context reasoning difficulties, we categorize questions in FRAMES benchmark into three groups—Low ($\leq$3), Medium (4–6), and High ($\geq$7)—and report the performance of LONGPAS on different reasoning depths. As shown in Figure 4, across all settings, both RLVR and LONGPAS improve substantially over the vanilla models; however, the advantage of LONGPAS becomes particularly pronounced as the reasoning complexity increases. For Qwen3-4B, while the vanilla model's performance drops markedly from 47.2% on low-hop questions to 36.8% on high-hop ones, LONGPAS maintains a much stronger performance, achieving 56.2% on high-complexity questions—an improvement of nearly 20 percentage points over the baseline. This gain is larger than that of RLVR (52.1%), indicating that LONGPAS is especially effective at stabilizing and enhancing the model's ability to handle long reasoning chains.

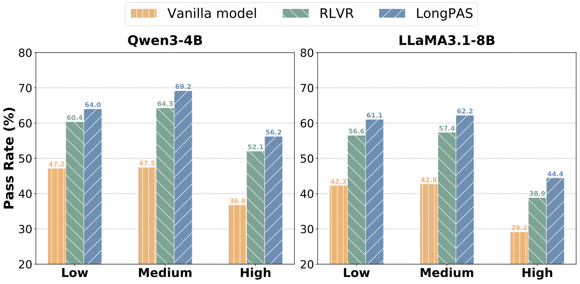


Figure 4: Performance of LONGPAS on FRAMES with different hop numbers. Questions are categorized into three complexities according to hop numbers: Low ($\leq$3), Medium (4-6) and High ($\geq$7).

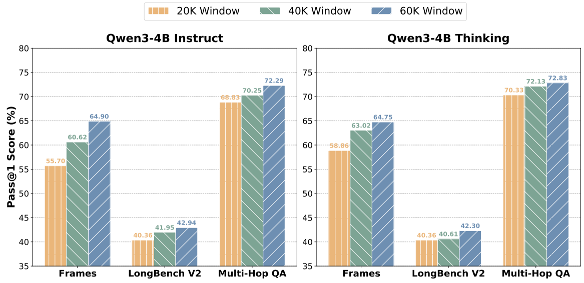


Figure 5: Performance of LONGPAS trained on different context windows (20K, 40K & 60K).

6.2 Ablation Study

In this section, we conduct ablation studies to investigate the key components in LONGPAS, including (1) the training data length; (2) process signals in advantage estimation; and (3) the role of on-policy trajectory supervision.

Training Data Length We investigate the impact of training data length on the performance of LONGPAS. Specifically, we train Qwen3-4B with varying maximum input lengths—20K, 40K, and 60K tokens—aligned with different context window sizes. As shown in Figure 5, the performance of the LONGPAS model, across both Qwen3-4B-Instruct and Qwen3-4B-Thinking variants, exhibits a monotonic increase with the expansion of the maximum input length during training, utilizing 20K, 40K, and 60K token context windows. This observation underscores the critical role of long-sequence training in improving the model's long-context grounding and reasoning. Specifically, the FRAMES and Multi-Hop QA tasks register the most substantial performance gains. This indicates that training with extended context windows significantly enhances the model's capability for complex information grounding and long-range dependency modeling, which are prerequisites for multi-step reasoning. While performance gains on LongBench V2 benchmark are less pronounced, the consistent positive correlation confirms that greater exposure to longer contexts during training systematically elevates the model's overall proficiency across diverse long-context tasks.

Process Signal of Advantage Shaping To disentangle the contributions of *Validity* and *Relevance* to process advantage shaping, we first remove the *Validity* signal $\mathbb{I}_{\text{valid}}(s_{i,j}) = \text{LLM}_{Judge}(s_j, \tau_p)$, and retaining only *Relevance* as the indicator of the necessity of sub-steps. Table 3 shows that excluding *Validity* results in a 2.22% and 1.79% drop on FRAMES and LongBench V2, respectively, highlighting the necessity of logical verification. Similarly, removing *Relevance* (binarizing the reweighting coefficient) leads to declines of 1.73% on FRAMES and 2.04% on Multi-Hop QA. These results confirm that both signals are essential: *Validity* ensures logical soundness, while *Relevance* filters contextual noise, together enabling precise credit assignment for intermediate reasoning steps.

On-policy Trajectory Supervision To evaluate the role of on-policy supervision, we conduct an ablation by replacing τ_p with off-policy trajectories τ_d sampled from Gemini-2.5-Pro under the guidance of GT reasoning chain. As shown in Table 3, we can also observe a performance decline across all benchmarks, notably on FRAMES (-2.01%). We attribute this to: (1) *Distribution Mismatch*: While τ_p aligns with the current policy's capability range, the off-policy τ_d from a superior teacher model is out-of-distribution. This forces the student to estimate advantages for reasoning paths it cannot yet explore, leading to unstable optimization. (2) *Static Complexity*: Teacher-generated steps often exhibit reasoning patterns or logic structures too complex for the student to emulate. Unlike the dynamic τ_p, the static nature of τ_d fails to synchronize with the policy's evolving learning progress.

6.3 Analysis

Training Dynamics To understand how LONGPAS achieves strong performance, Figure 6 illustrates the training dynamics of LONGPAS. Compared to GRPO and DAPO, LONGPAS exhibits markedly more stable optimization. Regarding the entropy curves, it is evidenced that LONGPAS preserves correct intermediate steps via fine-grained credit assignment, maintaining stable en-

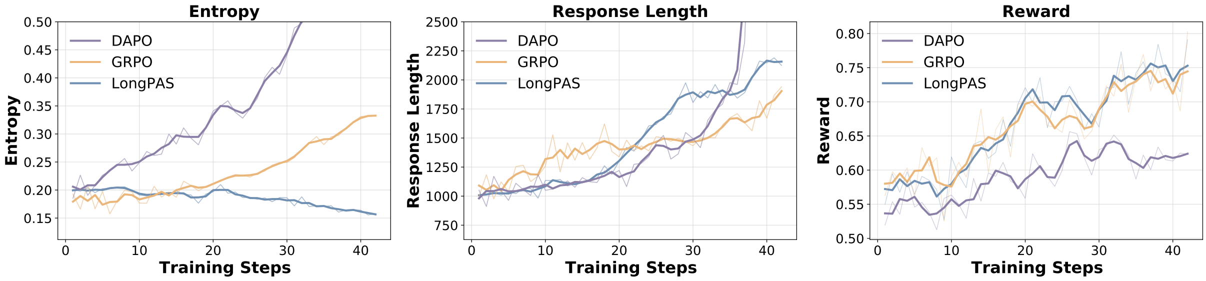


Figure 6: Training dynamics of LONGPAS on Qwen3-4B model compared with baseline algorithms. **Left**: Generation Entropy; **Middle**: Response Length; **Right**: Training Reward.

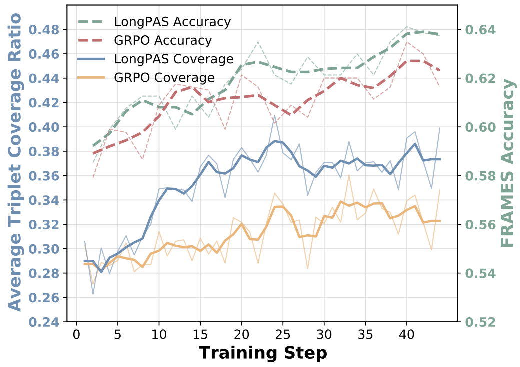


Figure 7: Triplet Coverage on the training data and FRAMES Accuracy dynamics with Qwen-4B model.

tropy throughout. Regarding the response length, LONGPAS maintains conciseness in early stages and increases length only when aligned with performance gains, effectively preventing verbosity exploitation. Conversely, DAPO shows unchecked length growth, eventually collapsing into meaningless token generation. Considering training reward, although LONGPAS initially lags behind GRPO, its reward curve maintains a consistent upward trend, ultimately surpassing GRPO and demonstrating superior long-term optimization. This confirms LONGPAS's efficacy in maximizing rewards through a more precise and stable learning signal.

Triplet Coverage Dynamics To elucidate how LONGPAS mitigates misapplied credit for "almost-there" samples, we analyze the relationship between average Triplet Coverage and accuracy (Entity Coverage Dynamics are shown in Appendix C.5). As shown in Figure 7, beginning from the same start, LONGPAS quickly establishes a superior Triplet Coverage (0.37–0.38), significantly outperforming GRPO. This sustained advantage suggests that LONGPAS enhances reasoning density by reinforcing valid logical steps. The synchronized growth of Triplet Coverage and accuracy confirms that grounding the reasoning process in structured evidence facilitates precise credit assignment. By incentivizing key triplet identification, LONGPAS effectively yielding more logical, evidence-based trajectories and superior performance.

7 Conclusion

In this work, we systematically investigate the "almost-there" phenomenon in long-context RL, highlighting how outcome-based RLCR overlooks critical learning signals in partially correct trajectories. We propose a KG-driven framework to synthesize **DEEPREASONQA**, a high-quality multi-hop QA dataset with explicit reasoning chains. We further introduce **LONGPAS**, which enables fine-grained credit assignment via process advantage shaping. Experimental results demonstrate that our approach significantly enhances long-context reasoning and enables smaller models to rival frontier LLMs, confirming that our approach stabilizes RL training and fosters complex logical integration. By integrating structured synthesis with granular process supervision, this paradigm provides a scalable path for developing agents capable of navigating complex, large-scale information environments.

Limitations

This paper aims to incentivize in-depth reasoning of long-context LLMs on long-range multi-hop QA tasks, and offers a recipe combining a KG-guided multi-hop QA synthesis framework with a process advantage shaping strategy. While it makes substantial progress in long-context reasoning, several limitations remain:

Data Domain and Source The synthesis framework currently relies primarily on Wikipedia as the underlying knowledge source. While DEEPREASONQA already yields substantial improvements and effectively incentivizes deep long-context reasoning patterns, incorporating long documents from domains such as law, finance, and medicine could introduce richer stylistic and structural diversity, potentially improving robustness and transferability to real-world long-context reasoning tasks.

Coupling of Synthesis Framework and Process Supervision A potential limitation is the perceived coupling between LONGPAS and the KG-driven synthesis framework, as the algorithm currently utilizes the specific reasoning chains generated during synthesis as the primary source of process signals. While this might appear to restrict the scalability of LONGPAS to datasets where explicit ground-truth reasoning paths are available , we argue that the core contribution of LONGPAS lies in its generalizable mechanism for Process Advantage Shaping. Specifically, the method provides a robust framework for effectively leveraging any form of auxiliary supervision—whether they are KG-derived reasoning chains or standard reasoning trajectories distilled from teacher LLMs—to stabilize long-context RL training. By transforming these signals into fine-grained credit assignment, LONGPAS addresses the fundamental "almost-there" bottleneck in a way that aligns with contemporary efforts (Deng et al., 2025; Zhu et al., 2025b) to integrate SFT-level supervision into the RL stage. Thus, rather than being a restricted implementation, our approach offers a novel and versatile perspective on mitigating reward sparsity in complex, long-context scenarios.

Sophistication of the Reward Model Our current implementation uses a hybrid reward function that combines simple rule-based checks with an LLM-as-a-judge to balance precision and recall. This works well for tasks with clear correctness criteria, such as factoid multi-hop questions, but may be less effective for open-ended or subjective tasks where correctness is multifaceted. A promising direction is to develop more advanced, rubric-based LLM reward models that score responses along dimensions such as logical rigor, citation accuracy, and coherence, enabling our framework to better handle complex, agentic scenarios.

Acknowledgments

This research was supported by the Guangdong Provincial Talent Program (No.2024TQ08X366). We would like to thank all the anonymous reviewers and area chairs for their insightful and valuable comments.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Sanghwan Bae, Jiwoo Hong, Min Young Lee, Hanbyul Kim, JeongYeon Nam, and Donghyun Kwak. 2025. Online difficulty filtering for reasoning oriented reinforcement learning. *CoRR*, abs/2504.03380.

Yushi Bai, Xin Lv, Jiajie Zhang, Yuze He, Ji Qi, Lei Hou, Jie Tang, Yuxiao Dong, and Juanzi Li. 2024a. Longalign: A recipe for long context alignment of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1376–1395.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024b. Longbench: A bilingual, multitask benchmark for long context understanding. In *ACL (1)*, pages 3119–3137. Association for Computational Linguistics.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *ACL (1)*, pages 3639–3664. Association for Computational Linguistics.

Guanzheng Chen, Xin Li, Michael Shieh, and Lidong Bing. a. Longpo: Long context self-evolution of large language models through short-to-long preference optimization. In *The Thirteenth International Conference on Learning Representations*.

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

Nuo Chen, Yuhan Li, Jianheng Tang, and Jia Li. 2024. Graphwiz: An instruction-following language model for graph computational problems. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 353–364.

Daixuan Cheng, Shaohan Huang, Xuekai Zhu, Bo Dai, Wayne Xin Zhao, Zhenliang Zhang, and Furu Wei. 2025. Reasoning with exploration: An entropy perspective. *arXiv preprint arXiv:2506.14758*.

Deepmind. 2025. Gemini 2.5: Our most intelligent ai model. Technical report, Deepmind.

Yihe Deng, I Hsu, Jun Yan, Zifeng Wang, Rujun Han, Gufeng Zhang, Yanfei Chen, Wei Wang, Tomas

Pfister, Chen-Yu Lee, and 1 others. 2025. Supervised reinforcement learning: From expert trajectories to step-wise reasoning. *arXiv preprint arXiv:2510.25992*.

Chaochen Gao, W Xing, Qi Fu, and Songlin Hu. a. Quest: Query-centric data synthesis approach for long-context scaling of large language model. In *The Thirteenth International Conference on Learning Representations*.

Chaochen Gao, W Xing, Zijia Lin, Debing Zhang, and Songlin Hu. b. Nextlong: Toward effective long-context training without long documents. In *Forty-second International Conference on Machine Learning*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? In *First Conference on Language Modeling*.

Zihao Jiang, Ben Liu, Miao Peng, Wenjie Xu, Yao Xiao, Zhenyan Shan, and Min Peng. 2025. Towards explainable temporal reasoning in large language models: A structure-aware generative framework. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7232–7251.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2025. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. In *NAACL (Long Papers)*, pages 4745–4759. Association for Computational Linguistics.

Huayang Li, Pat Verga, Priyanka Sen, Bowen Yang, Vijay Viswanathan, Patrick Lewis, Taro Watanabe, and Yixuan Su. 2024a. Alr2: A retrieve-then-reason framework for long-context question answering. *arXiv preprint arXiv:2410.03227*.

Jiaxi Li, Xingxing Zhang, Xun Wang, Xiaolong Huang, Li Dong, Liang Wang, Si-Qing Chen, Wei Lu, and Furu Wei. 2025. Wildlong: Synthesizing realistic long-context instruction data at scale. *arXiv preprint arXiv:2502.16684*.

Siheng Li, Cheng Yang, Zesen Cheng, Lemao Liu, Mo Yu, Yujiu Yang, and Wai Lam. 2024b. Large language models can self-improve in long-context reasoning. *arXiv preprint arXiv:2411.08147*.

Yanyang Li, Shuo Liang, Michael Lyu, and Liwei Wang. 2024c. Making long-context language models better multi-hop reasoners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2462–2475.

Zhan Ling, Kang Liu, Kai Yan, Yifan Yang, Weijian Lin, Ting-Han Fan, Lingfeng Shen, Zhengyin Du, and Jiecao Chen. 2025. Longreason: A synthetic long-context reasoning benchmark via context expansion. *arXiv preprint arXiv:2501.15089*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Belinda Mo, Kyssen Yu, Joshua Kazdan, Joan Cabezas, Proud Mpala, Lisa Yu, Chris Cundy, Charilaos Kanatsoulis, and Sanmi Koyejo. 2025. Kggen: Extracting knowledge graphs from plain text with language models. *Preprint*, arXiv:2502.09956.

OpenAI. 2025. Introducing gpt-5.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*.

Miao Peng, Nuo Chen, Zongrui Suo, and Jia Li. 2025. Rewarding graph reasoning process makes llms more generalized reasoners. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 2257–2268.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Weizhou Shen, Ziyi Yang, Chenliang Li, Zhiyuan Lu, Miao Peng, Huashan Sun, Yingcheng Shi, Shengyi Liao, Shaopeng Lai, Bo Zhang, and 1 others. 2025. Qwenlong-l1. 5: Post-training recipe for long-context reasoning and memory management. *arXiv preprint arXiv:2512.12967*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024a. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024b. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Huashan Sun, Shengyi Liao, Yansen Han, Yu Bai, Yang Gao, Cheng Fu, Weizhou Shen, Fanqi Wan, Ming Yan, Ji Zhang, and 1 others. 2025. Solopo: Unlocking long-context capabilities in llms via short-to-long preference optimization. *arXiv preprint arXiv:2505.11166*.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, and 1 others. 2025. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Fanqi Wan, Weizhou Shen, Shengyi Liao, Yingcheng Shi, Chenliang Li, Ziyi Yang, Ji Zhang, Fei Huang, Jingren Zhou, and Ming Yan. 2025. Qwenlong-l1: Towards long-context large reasoning models with reinforcement learning. *arXiv preprint arXiv:2505.17667*.

Siyuan Wang, Gaokai Zhang, Li Lyna Zhang, Ning Shang, Fan Yang, Dongyao Chen, and Mao Yang. 2025. Loongrl: Reinforcement learning for advanced reasoning over long contexts. *arXiv preprint arXiv:2510.19363*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Cehao Yang, Xueyuan Lin, Chengjin Xu, Xuhui Jiang, Shengjie Ma, Aofan Liu, Hui Xiong, and Jian Guo. 2025b. LongFaith: Enhancing long-context reasoning in LLMs with faithful synthetic data. In *Findings of the Association for Computational Linguistics: ACL 2025*, Vienna, Austria. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. 2025. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*.

Qifan Zhang, Nuo Chen, Zehua Li, Miao Peng, Jing Tang, and Jia Li. Improving llms 'generalized reasoning abilities by graph problems. In *Second Conference on Language Modeling*.

Zhiyang Zhang, Ziqiang Liu, Huiming Wang, Renke Shan, Li Kuang, Lu Wang, and De Wen Soh. 2025. Re3syn: A dependency-based data synthesis framework for long-context post-training. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31468–31480.

Yida Zhao, Kuan Li, Xixi Wu, Liwen Zhang, Dingchu Zhang, Baixuan Li, Maojia Song, Zhuo Chen, Chenxi Wang, Xinyu Wang, and 1 others. 2025. Repurposing synthetic data for fine-grained search agent supervision. *arXiv preprint arXiv:2510.24694*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Wenhao Zhu, Pinzhen Chen, Hanxu Hu, Shujian Huang, Fei Yuan, Jiajun Chen, and Alexandra Birch. 2025a. Generalizing from short to long: Effective data synthesis for long-context instruction tuning. *arXiv preprint arXiv:2502.15592*.

Wenqiao Zhu, Ji Liu, Rongjunchen Zhang, Haipang Wu, and Yulun Zhang. 2025b. Carft: Boosting llm reasoning via contrastive learning with annotated chain-of-thought-based reinforced fine-tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 5933–5948.

A Synthetic Data Details

A.1 Dataset Statistics

During data collection, we strictly precluded any overlap between the crawled documents and the test set to prevent data contamination. During the multi-hop QA synthesis pipeline in Section 4.2, we employ KGGen (Mo et al., 2025) to extract entities and triplets from plain texts. We sample reasoning paths from 2 to 30 hops to control the question difficulties. In "Question Generation" stage , we employ Gemini2.5-Pro (Deepmind, 2025) as the generator and Deepseek-V3-0528 (Liu et al., 2024) as the verifier to filter out questions with false answers. As a result, we construct the raw DEEPREASONQA dataset containing 14,577 QA samples with documents up to 288K tokens in length. Before RL training, we filter out trivial and unsolvable questions. For each question, we run 8 full-context rollouts with Qwen3-4B-Thinking and retain only those with an empirical success rate in [0.25,0.75], resulting in 2,012 QA samples with documents up to 60K tokens.

Table 4 presents the detailed data statistics for DEEPREASONQA, providing detailed information on the Raw Synthesis Dataset and the Filtered RL Training Dataset for better understanding. Figure 8 displays the distribution of QA categories in raw DEEPREASONQA. For readability, only categories with the top 40 frequency are displayed. In Figure 9, we show the word cloud of questions in raw DEEPREASONQA.

Furthermore, we summarize the data distribution of DEEPREASONQA in Figure 10. Figure 10a illustrates the token-length distribution of our training samples, which spans from 20K to 60K. Figure 10b shows the hop-number distribution, revealing that questions produced by our KG-based QA synthesis framework are highly challenging, with reasoning hops ranging widely from 2 to 30.

A.2 Detailed Quality Control Pipeline

Ideally, the constructed long-context multi-hop QAs should be well-grounded in supporting documents, with concise answer and high-quality reasoning paths. Thus, we applied a four-stage filtering pipeline to ensure the final DEEPREASONQA is of high quality:

Answer Alignment Check: During question generation, we use Gemini-2.5-Pro (Deepmind, 2025) to produce QA pairs based on task-specific prompts, documents, and reasoning paths. DeepSeek-V3-2508 (Liu et al., 2024) is then used to answer the generated questions, and GPT-OSS-120B (Agarwal et al., 2025) serves as a verifier to assess whether the two answers are consistent. Samples with misaligned answers are removed.

Knowledge Grounding Check: To reduce the potential bias from internal inherent knowledge, we temporarily remove the source documents and check whether the model can still answer correctly. Samples that remain answerable without the documents are filtered out to ensure the dataset genuinely tests contextual reasoning.

Complex Answer Filtering: QA pairs whose answers exceed 20 words are discarded, as overly complex answers are unstable and difficult to verify reliably.

Contextual Robustness Check: We augment each context with irrelevant documents and re-evaluate the model's answer. Samples whose answer accuracy (pass@k) drops to zero are removed, ensuring that each question–answer pair is robust rather than brittle under context perturbations.

A.3 Rationale behind the taxonomy of reasoning paradigms

We identify that high-reasoning-density questions necessitate the ability to synthesize deep and widely scattered contextual information for long-range reasoning. To this end, we categorize deep reasoning in long-context scenarios into following distinct paradigms:

- **In-depth Reasoning**: It includes **Multi-hop Reasoning** and **Causal Analysis**, which focus on tracking intricate logical chains across massive contexts to identify non-obvious dependencies and root causes.

- **Temporal Reasoning**: It requires aggregating discrete quantitative data spread throughout the text to perform precise calculations and model dynamic shifts over time.

- **Hypothetical Scenario**: It evaluates the ability of counterfactual reasoning by mapping existing logic onto new, speculative frameworks.

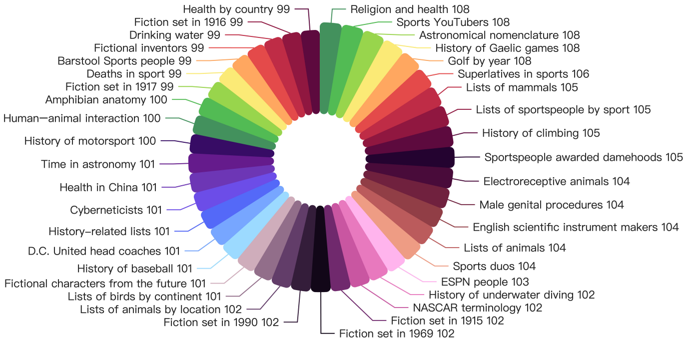


Figure 8: Category Distribution of DEEPREASONQA

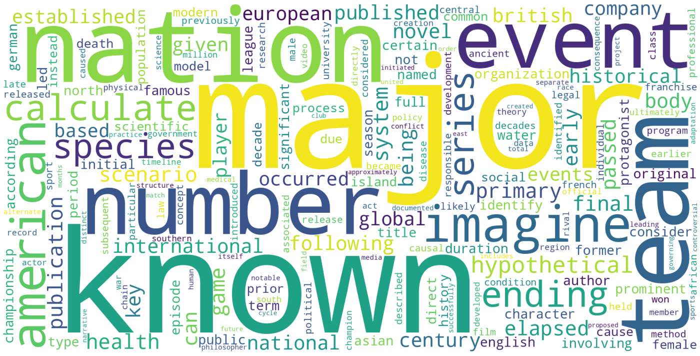
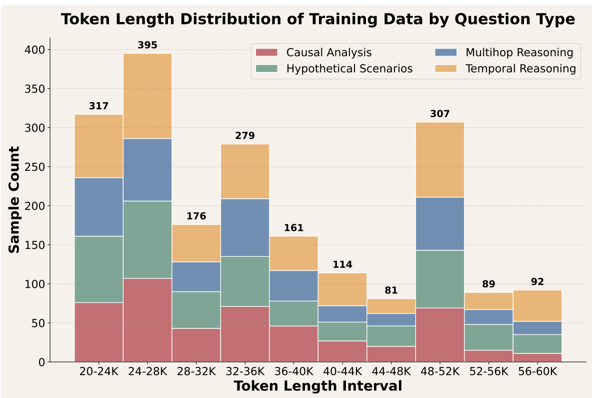


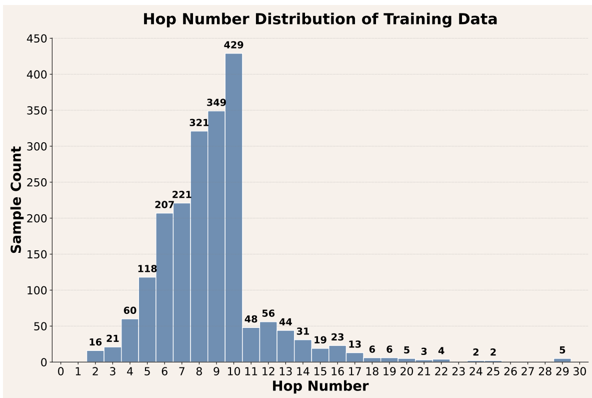
Figure 9: Question Word Cloud of DEEPREASONQA

Property	Number
Raw Synthesis Dataset	
# of Topic Categories	158
# of Total QA Pairs	14,577
# of Multi-hop Reasoning Question	4,415
# of Temporal Reasoning Question	4,345
# of Hypothetical Scenarios Question	3,137
# of Causal Analysis Question	2,680
# of Average Hop Number	8.78
# of Minimum Token Number	10,137
# of Maximum Token Number	288,515
# of Average Token Number	44,703
Filtered RL Training Dataset	
# of Total QA Pairs	2,012
# of Multi-hop Reasoning Question	447
# of Temporal Reasoning Question	572
# of Hypothetical Scenarios Question	508
# of Causal Analysis Question	485
# of Average Hop Number	8.78
# of Minimum Token Number	20,000
# of Maximum Token Number	59,888
# of Average Token Number	35,876

Table 4: Data Statistics of DEEPREASONQA



(a) Token Length Distribution



(b) Hop Number Distribution

Figure 10: Distributions of token length (a) and hop numbers (b) in DEEPREASONQA. Texts are tokenized by Qwen3-4B-2507 tokenizer.

B More Experimental Details

Benchmark Details In our experiments, we evaluate the long-context capabilities of our model and baselines on several widely-used benchmarks:

- **FRAMES** (Krishna et al., 2025): FRAMES is a comprehensive evaluation dataset designed to test the capabilities of LLMs across factuality, retrieval accuracy, and reasoning. It contains 824 complex, multi-hop reasoning questions related to a diverse set of topics from Wikipedia, involving subjects such as history, sports, science, animals, health, etc. Each question is labeled based on five reasoning types (i.e, Numerical Reasoning, Tabular Reasoning, Multiple Constraints, Temporal Reasoning, and Post-Processing), which requires 2-15 Wikipedia articles to answer. To align with our long-context setting, we equip each question with corresponding wikipedia pages to minimize the effect of information retrieval.

- **LongBench V2** (Bai et al., 2025): LongBench V2 is a benchmark designed to evaluate LLMs' ability to tackle long-context tasks that require deep comprehension and multi-step reasoning across real-world scenarios. It comprises 503 challenging multiple-choice questions with context lengths ranging from 8K to 2M words, covering six key task categories: single-document QA, multi-document QA, long in-context learning, long-dialogue understanding, code repository comprehension, and long structured-data understanding.

- **Multi-Hop QA** (Bai et al., 2024b): Multi-Hop QA consist of three subsets including 2WikiMultiHopQA (Ho et al., 2020), HotpotQA (Yang et al., 2018) and MusiQue (Trivedi et al., 2022) that are adopted from LongBench (Bai et al., 2024b). HotpotQA, 2WikiMultiHopQA, and MuSiQue are constructed among wikipedia or wikidata, via different multi-hop mining strategies with crowd-sourcing. They cover 3-5 hop questions with corresponding documents.

Implementation Details During Process Advantage Shaping in Section 4.3, we adopt GPT-OSS-120B as the LLM-as-Judge for answer evaluator and provide *Validity* signals in Process Advantage Estimation. We use Qwen3-8B-Embedding to calculate semantic similarity between trajectory steps. For Thinking models, since their outputs contain "thinking trajectories" enclosed by "<think>" and "</think>" tokens, we assign the average advantage of the "response" part uniformly to each token within this thinking segment.

SFT Configurations To construct the SFT dataset, we adopt DEEPREASONQA (2k QA pairs) and employ DeepSeek-V3-0528 and DeepSeek-R1-0528 to generate teacher trajectories under the guidance of ground-truth reasoning chains, for the Instruct model and Thinking model respectively. We filter out samples with incorrect final answers and retain 1,536 instances. The input length in the SFT stage is set to 60K. Training is conducted for 4 epochs with a batch size of 256 and a learning rate of 1e-5.

Evaluation Configurations For all LLM backbones, we conduct evaluation under a 128K context window. Specifically, LLaMA3.1-8B-Instruct and Qwen2.5-7B-Instruct (with YaRN enabled) are evaluated with a maximum input length of 120K and an output limit of 10K. For all frontier models and the Qwen3 series, the input length is also set to 120K, with output limits of 10K for Instruct models and 30K for reasoning models. For each question, we generate $N = 4$ candidate responses and report the average score (Pass@1) in our main experiments, as well as Pass@k for test-time scaling analyses. The Pass@k metric provides an unbiased estimate of the probability that at least one of the k sampled responses is correct, given n candidate solutions per problem. For multiple-choice tasks, we report standard accuracy. For open-end multi-hop QA tasks, we use the Hybrid Reward mentioned in Section 4.3 and use GPT-OSS-120B as LLM-as-a-judge (Zheng et al., 2023) to evaluate semantic equivalence between a model's prediction and the ground-truth answer.

Training Prompt We list the training prompt template we used during training in **Prompt 1**.

LLM-as-judge Prompt We list the prompts of LLM-as-Judge (1) when used as Outcome Reward Model to judge whether predicted answer is aligned with ground-truth answer (**Prompt 2**); (2) when used as Sub-step Validity Signal to judge whether a sub-step in rollout trajectories aligns with the necessary entities and reasoning logic of the reference trajectory (**Prompt 3**).

C Further Analysis

C.1 Generalization on Longer Context Window

Although LONGPAS is trained on a 60K input context window, we observe a strong generalization capacity to much longer contexts. As shown in Table 5, LONGPAS achieve the strongest gains in the 16K–64K range—closest to the 60K training length. Notably, this improvement remains substantial even beyond 64K context window, while other baselines degrade sharply: LONGPAS lifts LLaMA3.1-8B-Instruct from 34.62 to 48.08 and Qwen3-4B-Instruct from 51.92 to 57.69 on FRAMES with >64K input contexts. Upon questions with more than 128K contexts in Longbench V2, they achieve impressive absolute gains of +4.63% and +6.48%, respectively. Meanwhile, LONGPAS exhibits strong performance on shorter

Prompt 1: System Prompt during Training

You are a helpful assistant. Please read the provided text and answer the question below. Please structure your response into two main sections: Thought and Solution. In the Thought section, detail your reasoning process using the specified format: `<begin_of_thought>` thought with steps start with 'Step N:' `<end_of_thought>` Each step should include detailed considerations such as analyzing questions, summarizing relevant findings, brainstorming new ideas, verifying the accuracy of the current steps, refining any errors, and revisiting previous steps. In the Solution section, based on various attempts, explorations, and reflections from the Thought section, systematically present the final solution that you deem correct. The solution should remain a logical, accurate, concise expression style and detail necessary step needed to reach the conclusion, formatted as follows: `<begin_of_solution>` Therefore, the answer is {insert answer here} `<end_of_solution>`.

Prompt 2: Prompt for LLM-as-Judge as Reward Model

You are an expert in verifying if two answers are the same.
Your input is a problem and two answers, Answer 1 and Answer 2. You need to check if they are equivalent.
Your task is to determine if two answers are equivalent, without attempting to solve the original problem.
Compare the answers to verify they represent identical values or meaning, even when written in different forms or notations.

Your output must follow the following format:
1) Provide an explanation for why the answers are equivalent or not.
2) Then provide your final answer in the form of: [[YES]] or [[NO]]

Problem: {question}
Answer 1: {predicted answer}
Answer 2: {golden answer}

input contexts (<16K). These results show that LONGPAS not only attains optimal performance at its trained context length but also generalizes robustly to longer contexts, particularly on complex multi-hop reasoning tasks.

C.2 Analysis on the Quality of GT Reasoning Chains

During the Long-Context QA Synthesis stage, we obtain multi-hop QA pairs together with their corresponding multi-hop reasoning chains. In this section, we validate the profound impact of providing explicit Ground Truth reasoning chains on solving high-difficulty multi-hop QA problems, which were previously unsolvable even after 8 rollouts (Pass@8 = 0). Figure 11 clearly shows that the GT-guided prompts enabled LLMs to solve up to 69.4% of these hard samples, confirming the effectiveness of this "step-level supervision" in long-text reasoning. Specifically, the combination of GT reasoning chains on Thinking model gains greater improvement, achieving a Pass@1 success rate of 43.9% and reducing the total failure rate to 30.6%. This demonstrates that the explicit reasoning chains effectively improve LLMs' answer accuracy on long-context multi-hop reasoning tasks, even on extremely hard questions, allowing the LLM to effectively trace the complex, multi-hop logic that connects the question to the reference answer.

C.3 Observation of "almost-there" phenomenon on training datasets

In Section 4.1, we analyzed the commonly observed "almost-there" phenomenon during RL training, where trajectories are largely correct but fail at the end due to a minor error. In this sec-

Prompt 3: Prompt for LLM-as-Judge as Sub-step Validity Signal

You are an expert in analyzing reasoning traces. Your task is to determine if a "given reasoning substep from a model's output" is contained or reflected within the "Ground Truth reasoning solution".

In your assessment, you must strictly adhere to the following special rules:
1. Ignore Step Order: You need to check if the logical content or core reasoning expressed by the substep is covered by the Ground Truth path, regardless of its position in either path.
2. Accept Varied Granularity: Differences in reasoning granularity are allowed. If the model substep is a logical combination (merger) of multiple steps in the Ground Truth, or if it is only a part (subset) of a single Ground Truth step, it should still be considered a match, as long as its core logic is clearly included or reflected in the Ground Truth path.

Specifically, you should check:
1. Does the substep text or its semantic equivalent appear in the Ground Truth solution?
2. Is the substep's core logic or reasoning step reflected or contained within the Ground Truth solution?
3. Does the substep represent a logical component that exists within the Ground Truth reasoning process?

You are checking if the substep EXISTS in the Ground Truth, not if it's correct or necessary for solving the problem.
Provide your final answer in the form of:
[[YES]] or [[NO]]

Ground Truth Reasoning Solution: {ground truth}
Reasoning Substep to Check: {substep}

Models	FRAMES				LongBench V2		
	0-16K	16K-32K	32K-64K	>64K	Short	Medium	Long
LLaMA3.1-8B-Instruct	41.25	43.60	41.18	34.62	31.67	**27.21**	23.15
- RLVR	57.34	55.69	52.35	34.62	34.31	25.93	25.46
- **LONGPAS**	**60.62**	**60.57**	**62.65**	**48.08**	**34.86**	26.16	**27.78**
Qwen3-4B-Instruct	47.60	45.93	44.12	51.92	44.44	34.30	31.25
- RLVR	59.48	63.72	62.35	53.85	47.36	**37.21**	36.81
- **LONGPAS**	**63.39**	**67.68**	**66.47**	**57.69**	**53.47**	36.74	**37.73**
Qwen3-4B-Thinking	60.18	60.37	**66.76**	55.77	47.64	35.35	35.42
- RLVR	55.83	61.38	58.82	53.85	50.00	35.81	**39.81**
- **LONGPAS**	**63.12**	**67.38**	66.47	**63.46**	**52.08**	**36.05**	38.43

Table 5: Overall **Pass@1** performance on long-context QA benchmarks. The top scores of each backbone LLM are **bolded**. Data in LongBench V2 is divided into three groups: Short (<32K), Medium (32K-128K), and Long (>128K).

tion, we further examine whether a similar Entity & Triplet coverage trend appears on DEEPREASONQA. As shown in Figure 12, both Entity Coverage and Triplet Coverage tend to increase as group accuracy improves. Moreover, the coverage scores peak and stay above the overall average when the

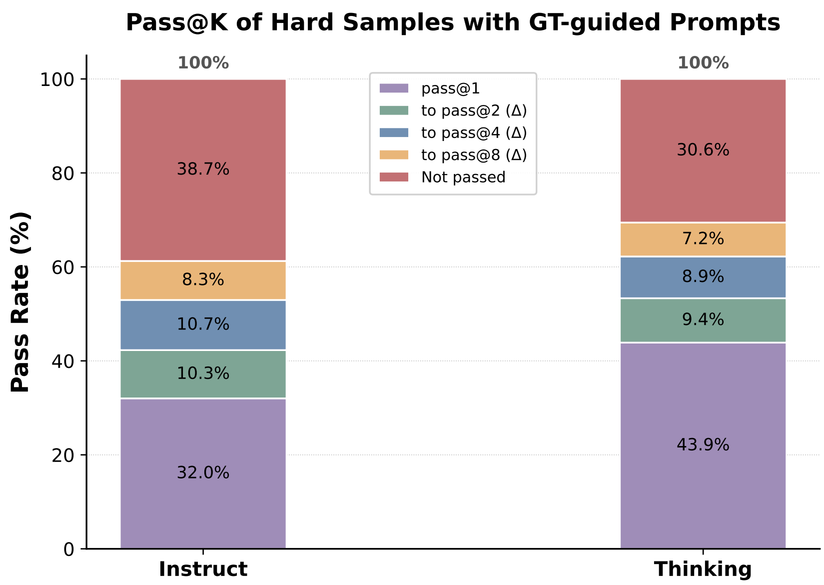


Figure 11: Pass@K performance of Qwen3-4B models on hard samples using GT-guided prompts. For each question in DEEPREASONQA, we generate 8 rollouts and retain only those samples for which all predictions are incorrect (accuracy = 0).

Group Positive Ratio is around 50–75%. This further demonstrates that LLMs achieve stronger information grounding but still fail at the subsequent combination and reasoning stage, underscoring the necessity of handling such "almost-there" cases during RL training.

C.4 Training Dynamics of Thinking Model

We further analyze the step-by-step training dynamics of Qwen3-4B-Thinking model to further understand the training behaviors of LONGPAS on Thinking model. As shown in Figure 13,

C.5 Entity Coverage Dynamics

To further investigate the mechanism of LONGPAS in mitigating the common wrong credit assignment confronting "almost-there" samples, we analyze the dynamics comparison between average Entity Coverage Ratio and FRAMES Accuracy during training period. As shown in Figure 14, although both methods start at the same level, LONGPAS quickly establishes and consistently maintains a clear advantage in the Average Entity Coverage Ratio. Its coverage curve stabilizes at a higher range around 0.60, notably above GRPO's curve, which oscillates around 0.56. This indicates that LONGPAS is more effective at grounding its reasoning in essential contextual information. Moreover, LONGPAS attains a consistently higher peak in FRAMES accuracy compared with GRPO, demonstrating that its step-level advantage shaping yields more precise credit assignment across the reasoning process, ultimately resulting in superior overall task performance.

C.6 Test Time Scaling

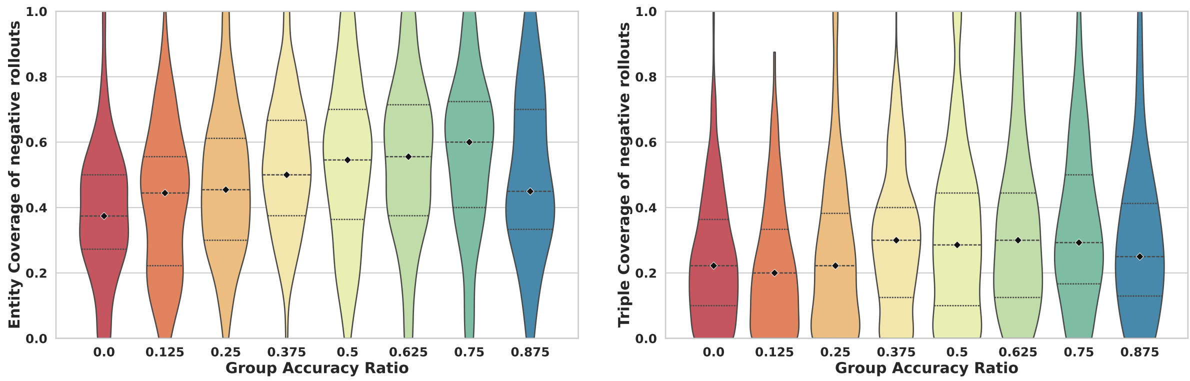
Prior studies have shown that with a limited number of rollouts, models often struggle to solve certain tasks, whereas a sufficiently large rollout budget substantially increases the probability of sampling effective solutions. Figure 15 reports the Pass@k performance of LONGPAS under test-time scaling settings. The results show that LONGPAS achieves consistent gains as k increases from 1 to 8. Notably, LONGPAS also attains a higher Pass@1 score than Vanilla-GRPO, highlighting its effectiveness in boosting LLMs to produce precise reasoning processes for complex long-context multi-hop tasks during RL training.

C.7 Detailed Results on LongBench V2

We report the detailed results on each sub-task of LongBench V2 in Table 6 to better illustrate the effectiveness of LONGPAS.

C.8 Case Study

To illustrate the qualitative differences in reasoning, we present a comparative case study using trajectories generated by Qwen3-4B-Thinking and LONGPAS for the same question.



(a) Entity Coverage

(b) Triplet Coverage

Figure 12: Entity Coverage (a) and Triplet Coverage (b) distribution between negative rollouts and ground-truth reasoning chains on DEEPREASONQA during training stage. Distributions are calculated according to average score in each group.

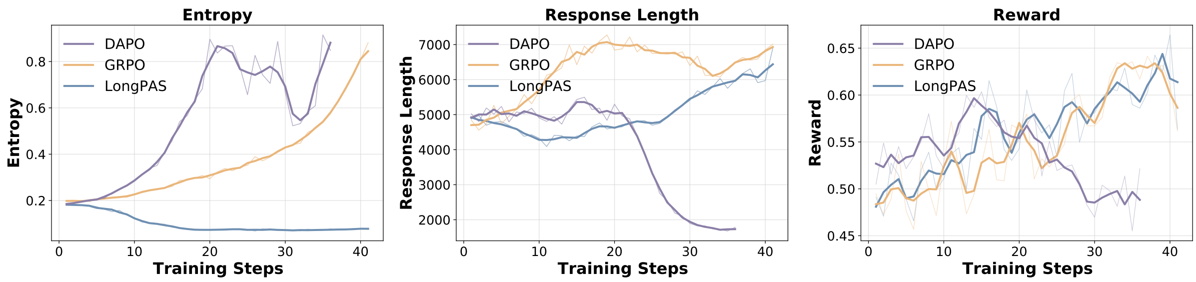


Figure 13: Training dynamics of LONGPAS on Qwen3-4B-Thinking model compared with baseline algorithms. **Left**: Generation Entropy; **Middle**: Response Length; **Right**: Training Reward.

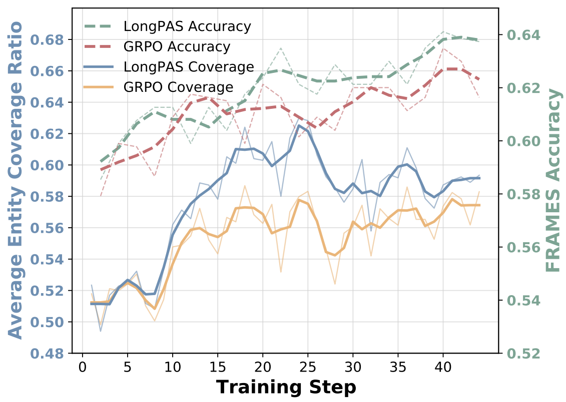


Figure 14: Average Entity Coverage Ratio (%) on the training data and FRAMES Accuracy dynamics with Qwen-4B model.

As shown in the examples below, the task involves multi-hop fact retrieval and date-based quantitative reasoning. The model must extract three pieces of information from different parts of the documents: (i) the person after whom the ship was named, (ii) the year the ship sank, and (iii) the birth and death years of that person, and then perform logical comparison and arithmetic calculation. The reasoning trajectory of LONGPAS exhibits more advanced critical reasoning and temporal logic, covering stages such as Information Grounding, Information Extraction, Strategy Adjustment, Temporal Calculation, Self-Correction, and Answer Confirmation.

Comparing the two reasoning trajectories, we observe a shift from "simple pattern matching and computation" to "logical reasoning based on state judgment". The reasoning trajectory of LONGPAS exhibits several richer reasoning patterns: (1) **Exhaustive Evidence Retrieval**: It performs more detailed and systematic information grounding, actively scanning and cross-checking relevant spans instead of relying on superficial matches; (2) **Constraint Consciousness**: It shows a stronger awareness of task boundaries and constraints throughout the thinking process; (3) **Robust Verification Loop**: It adopts a more divergent validation strategy, revisiting candidate answers, cross-validating them with multiple pieces of evidence, and rejecting inconsistent hypotheses before committing to a final conclusion.

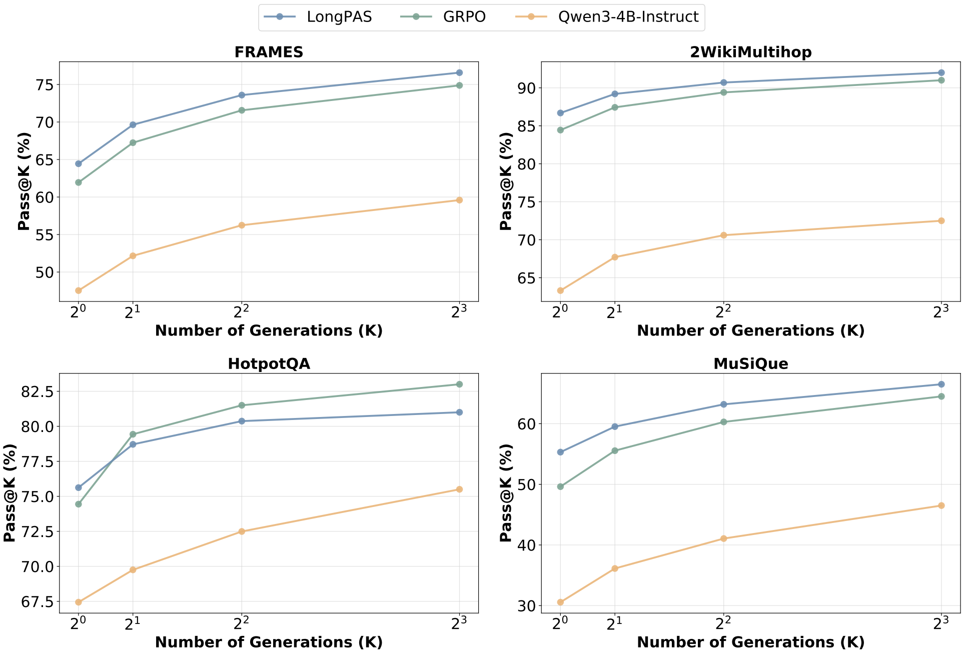


Figure 15: Test Time Scaling Performance (Pass@k) on four multi-hop reasoning benchmarks. The number of generations k varies from 1 to 8.

We also observe that both models share an initial segment of correct reasoning steps, but the vanilla model later diverges into incorrect deductions and calculations. This phenomenon further demonstrates that LONGPAS can better learn from "almost-there" samples during RL training, leading to more accurate and precise reasoning.

D Prompts & Cases

D.1 QA Generation Prompts

During question generation in Section 4.2, we employ specialized prompts to synthesize questions under different reasoning paradigms, including Multi-hop Reasoning, Temporal Reasoning, Causal Analysis and Hypothetical Scenarios. For better understanding, we showcase the detailed prompt used for multi-hop reasoning QA generation below:

D.2 QA Cases of DEEPREASONQA

In this section, we list the detailed cases of RL training data we constructed during Knowledge-Guided Long-Context Multi-hop QA Synthesis in Section 4.2, including Multi-hop Reasoning, Temporal Reasoning, Causal Analysis and Hypothetical Scenarios.

Models	SingleDoc	MultiDoc	Code Repo	Dialogue	Long ICL	Long SDU	**Overall**
Frontier Models							
GPT5-Nano	44.00	39.20	50.00	46.15	44.44	45.45	43.74
Gemini-2.5-Flash-Thinking	51.43	55.20	58.00	66.67	72.84	37.50	56.77
GPT-OSS-120B	44.57	43.20	53.06	61.54	46.91	48.48	47.01
GPT-OSS-20B	38.51	40.80	56.00	61.54	39.74	46.88	43.37
Instruct Models							
LLaMA3.1-8B-Instruct	28.86	26.20	27.00	32.05	26.23	30.30	27.93
- RLVR	31.86	29.20	29.00	32.69	23.46	21.21	28.93
- **LongPAS**	30.14	29.80	32.00	34.62	27.16	22.73	**29.62**
Qwen2.5-7B-Instruct	36.71	28.20	32.00	35.90	36.42	30.30	33.60
- RLVR	34.43	29.00	30.00	40.38	25.31	25.00	31.01
- **LongPAS**	39.57	28.40	29.50	31.41	35.49	27.27	**33.70**
Qwen3-4B-Instruct	36.00	30.60	36.50	60.26	41.67	32.58	37.28
- RLVR	39.14	37.00	49.50	60.90	43.21	40.91	42.10
- **LongPAS**	38.14	40.60	48.00	63.46	45.37	39.39	**42.94**
Qwen3-30B-A3B-Instruct	42.43	38.40	52.00	62.82	47.53	37.21	44.43
- RLVR	50.29	50.40	42.00	56.41	43.21	36.36	47.91
- **LongPAS**	46.29	42.40	56.00	71.79	53.09	42.42	**49.11**
Reasoning Models							
Qwen3-4B-Thinking	37.00	35.60	41.50	60.26	44.44	42.42	40.46
- RLVR	37.14	40.80	46.00	66.67	41.98	33.33	41.75
- **LongPAS**	40.29	40.00	43.00	62.82	40.43	40.91	**42.30**
Qwen3-30B-A3B-Thinking	44.00	44.00	46.00	64.10	55.56	54.55	48.31
- RLVR	40.00	46.40	50.00	51.28	49.38	33.33	44.53
- **LongPAS**	47.43	54.40	60.00	76.492	58.02	45.45	**54.27**

Table 6: Overall performance of models on LongBench V2 benchmark. RLVR is implemented with GRPO (Shao et al., 2024). Results are reported using the **Avg@4** metric. The top scores for each backbone LLM are **bolded**.

Case Trajectory on FRAMES with LongPAS-4B-Thinking (Part 1)

Question: If the man that the SS Edmund Fitzgerald was named after was living at the time of the ship's sinking, how old was he? If he was already deceased, how long had he been dead? You may just use the year without regard for the date of birth.

Ground Truth: Edmund Fitzgerald, the man for whom the ill-fated ship SS Edmund Fitzgerald was named was born in 1895, he was 80 years old in 1975 when she sank.

<think>
I need to answer the question about how old Edmund Fitzgerald would have been if he was alive at the time of the ship's sinking, or how long he had been dead if he was already deceased. The question says I can just use the year without regard for the date of birth.

First, I need to find out who Edmund Fitzgerald was and when he was born and died. From the text provided, I can see that the ship was named after someone named Edmund Fitzgerald.

Let me look through the text to find information about Edmund Fitzgerald (the person, not the ship).

In the "Name and launch" section, I find this information:

"Northwestern Mutual wanted to name the ship after its president and chairman of the board, Edmund Fitzgerald. Fitzgerald's own grandfather and all great uncles had themselves been lake captains, and his father owned the Milwaukee Drydock Company, which built and repaired ships. Fitzgerald had attempted to dissuade the naming of the ship after himself, proposing the names Centennial, Seaway, Milwaukee and Northwestern. The board was resolute, and Edmund abstained from voting; the 36 board members voted unanimously to name her the SS Edmund Fitzgerald."

This tells me that Edmund Fitzgerald was the president and chairman of the board of Northwestern Mutual Life Insurance Company. He was the person after whom the ship was named.

I need to find out his birth and death dates. The text doesn't explicitly state his birth and death dates. I might need to use external knowledge, but since this is a reading comprehension task, I should stick to the information provided in the text.

Let me search the entire text for any mention of Edmund Fitzgerald's age or death.

[Following thinking texts abbreviated]
</think>

Case Trajectory on FRAMES with LongPAS-4B-Thinking (Part 2)

<begin_of_thought>

Step 1: I need to identify who the SS Edmund Fitzgerald was named after. The question asks about "the man that the SS Edmund Fitzgerald was named after." From the provided text, in the "Name and launch" section, it states: "Northwestern Mutual wanted to name the ship after its president and chairman of the board, Edmund Fitzgerald." **This confirms that the ship was named after Edmund Fitzgerald**, who was the president and chairman of the board of Northwestern Mutual Life Insurance Company.

Step 2: The next step is to determine if this Edmund Fitzgerald was alive at the time of the ship's sinking (November 10, 1975) and, if so, his age or how long he had been dead. The question specifies that I can use just the year without regard for the date of birth, so I need to find his birth and death years from the text. I carefully scanned the entire provided text for any mentions of Edmund Fitzgerald's personal details, including age or lifespan.

Step 3: **In the "See also" section of the text, I found relevant information**: "Edmund B. Fitzgerald (1926–2013), American businessman, and son of the namesake (1895–1986) of the ship." This clearly states that the namesake (the person the ship was named after) was born in 1895 and died in 1986. The phrase "namesake (1895–1986)" directly indicates the birth and death years of the individual after whom the ship was named. This is the key information needed to answer the question, as it provides the exact years without ambiguity.

Step 4: Now I need to verify if Edmund Fitzgerald (the namesake) was alive during the ship's sinking in 1975. Since he died in 1986, **he was definitely alive in 1975** (as 1975 is before 1986). The question asks for his age if he was living at the time of sinking. Using the year-based calculation (as instructed to ignore dates of birth), I subtract his birth year from the sinking year: 1975 - 1895 = 80. The text specifies to use the year without regard for the date of birth, so I don't need to consider whether his birthday had occurred in 1975—this simplifies the calculation to 80 years old.

Step 5: I double-checked for any potential errors or alternative interpretations. The text confirms Edmund Fitzgerald was alive in 1958 (when the ship was launched, as he was involved in the naming process), and since he died in 1986, he was alive during the 1975 sinking. No other parts of the text mention his age or death date, so the "See also" section is the only reliable source here. The calculation 1975 - 1895 = 80 is straightforward and aligns with the question's instruction to use years only. There are no contradictions in the text, so this is accurate.

<end_of_thought>

<begin_of_solution>

Therefore, the answer is 80.

<end_of_solution>

Case Trajectory on FRAMES with vanilla Qwen3-4B-Thinking-2507 (Part 1)

Question: If the man that the SS Edmund Fitzgerald was named after was living at the time of the ship's sinking, how old was he? If he was already deceased, how long had he been dead? You may just use the year without regard for the date of birth.

Ground Truth: Edmund Fitzgerald, the man for whom the ill-fated ship SS Edmund Fitzgerald was named was born in 1895, he was 80 years old in 1975 when she sank.

<think>
Let me carefully analyze the text to find information about Edmund Fitzgerald (the person after whom the ship was named) and his age or death date.

From the provided text, I can find this relevant information in the "Name and launch" section:

"Northwestern Mutual wanted to name the ship after its president and chairman of the board, Edmund Fitzgerald. Fitzgerald's own grandfather and all great uncles had themselves been lake captains, and his father owned the Milwaukee Drydock Company, which built and repaired ships. Fitzgerald had attempted to dissuade the naming of the ship after himself, proposing the names Centennial, Seaway, Milwaukee and Northwestern. The board was resolute, and Edmund abstained from voting; the 36 board members voted unanimously to name her the SS Edmund Fitzgerald."

This tells me that Edmund Fitzgerald was the president and chairman of the board of Northwestern Mutual, and the ship was named after him.

I need to find out when Edmund Fitzgerald died (or if he was alive at the time of the sinking).

Let me look for information about Edmund Fitzgerald's death in the text.

I see a note at the end of the text that says: "Edmund Fitzgerald may also refer to:"

And it lists several people with the same name:

- "The Wreck of the Edmund Fitzgerald", a 1976 song about the shipwreck, by Gordon Lightfoot
- Edmund Fitzgerald Fredericks (1874/75–1935), Guyanese lawyer and Pan-African activist
- Edmund B. Fitzgerald (1926–2013), American businessman, and son of the namesake (1895–1986) of the ship
- The Edmund Fitzgerald (band), a UK math rock group

[Following thinking texts abbreviated]
</think>

Case Trajectory on FRAMES with vanilla Qwen3-4B-Thinking-2507 (Part 2)

<begin_of_thought>

Step 1: I need to identify who the SS Edmund Fitzgerald was named after and determine if he was alive at the time of the sinking (November 10, 1975) or had already died.

Step 2: From the text, I find that "Northwestern Mutual wanted to name the ship after its president and chairman of the board, Edmund Fitzgerald." This confirms that the ship was named after a man named Edmund Fitzgerald who was the president and chairman of the board of Northwestern Mutual.

Step 3: I need to find information about when this Edmund Fitzgerald died. Looking at the end of the text, I see a section that says: "Edmund B. Fitzgerald (1926–2013), American businessman, and son of the namesake (1895–1986) of the ship". This indicates that the ship's namesake (the man the ship was named after) was Edmund Fitzgerald who lived from 1895 to 1986.

Step 4: The ship sank on November 10, 1975. Since the ship's namesake died in 1986, he had already been dead for 1986 - 1975 = 11 years by the time of the sinking. [Wrongly deduce the conclusion that one is "Dead."]

Step 5: I verify that the note specifically states "the namesake (1895–1986) of the ship," which means the ship's namesake was Edmund Fitzgerald who died in 1986. There is no indication in the text that this person was alive at the time of the sinking. [Unsuccessful self-correction.]

Step 6: The question asks "If the man that the SS Edmund Fitzgerald was named after was living at the time of the ship's sinking, how old was he? If he was already deceased, how long had he been dead?" Since the man was deceased (died in 1986), and the ship sank in 1975, he had been dead for 11 years. [Miscalculation of years of death.]

Step 7: I confirm that the question says "You may just use the year without regard for the date of birth," so I don't need to calculate exact months or days. The calculation of 1986 - 1975 = 11 years is sufficient.

<end_of_thought>

<begin_of_solution>

Therefore, the answer is **11 years.**

<end_of_solution>

Example of Multi-hop Reasoning QA Generation Prompt

TASK: As a specialist in complex problem design, you will be provided with {k_context} Wikipedia articles and a set of "Multi-hop Reasoning Paths" derived from a knowledge graph. Your goal is to generate one challenging factoid "Multihop Reasoning" question that meets the following criteria.

Your design principles are to maximize reasoning steps, maximize information lookup dependencies, and minimize direct clues while ensuring uniqueness and verifiability of the answer.

1. Core Requirements

* Standalone & Context-Independent: Questions must not reference "Article 1," "Article 2," etc. They should be fully understandable without any additional context.
* Unambiguous & Verifiable Answer: The question should have a single, clear, and factual answer.
* Chain-of-Thought Reasoning: The question must be structured to force a rigorous, step-by-step reasoning process. Each logical step must build upon the previous one, following the provided multi-hop reasoning path.

2. Advanced Reasoning Requirements

* Multi-hop Reasoning via Knowledge Graph Paths: Each question must be constructed by tracing and combining information along the provided multi-hop reasoning path. The path acts as a logical blueprint, connecting entities and concepts from different articles. It's important to note that the path may only involve a subset of the {k_context} articles.
* Multi-hop Reasoning Path Format: (Subject 1)-[Relation 1]-(Subject 2)-[Relation 2]-(Subject 3)-[Relation 3]-(Subject 4)...
* Generate a problem that requires reasoning through multiple entities and relationships. The problem should call for starting from one entity, reaching another through multiple chains of relationships, and analyzing the significance of this connection.

3. Output Format

For QA pair, follow this exact format:
[[Question]]:
[[Answer]]:
[[Explanation]]: Clearly explain the reasoning process. For each step, bullet point the specific piece of information (including the number/fact and the article it came from) used from the Wikipedia articles to formulate the question and its answer.

[N In-context documents and reasoning paths demonstrations abbreviated]

Case of KG-guided Synthesis: **Multi-hop Reasoning**

Question:
What European capital served as the city of exile for the head of a six-generation publishing house, a central figure in a Los Angeles Times award-winning debut novel, who shares the narrative with a Canadian academic? This academic's research focus on nostalgia and subsequent mental decline thematically links to the professional specialization of a comatose doctor from a canonical comic book. This comic's narrative begins in the same year that a future 'Savior' in a fantasy television series, then a homeless youth in a Midwestern U.S. state, was inspired by reading 'The Ugly Duckling' to choose her surname.

Answer: Vienna.

Reasoning Chain:
(Emily Oliver)-[dismisses]-(Nádja)-[is a close friend of]-(John Price)-[is the protagonist of]-(Prague)-[deals with the history of]-(Horváth Kiadó)-[is the head of]-(Imre Horváth)-[was exiled in]-(Vienna)

Case of KG-guided Synthesis: **Hypothetical Scenario**

Question:
Imagine a hypothetical scenario where a warlord from the Muromachi-Azuchi period, whose martial philosophy was famously summarized as "being crazy to die," is tasked with analyzing the ethical underpinnings of the celebrated story of the masterless warriors of the AkÅ Domain. While acknowledging their loyalty, this 16th-century figure would likely find their actions to be a departure from the more pragmatic, victory-focused ethos of his own era. Based on the historical development of the samurai moral code, what philosophical system, which became a required norm for samurai for the first time during the subsequent era of prolonged peace, would he identify as the primary influence that reshaped the warrior's way into a more refined moral and ethical theory?

Answer: Confucianism.

Reasoning Chain:
(Chūshingura (A Treasury of Loyal Retainers))-[tells the story of]->(Forty-seven rōnin of the Akō Domain)-[were sentenced to]->(seppuku)-[is part of]->(Bushido)-[was influenced by]->(Confucianism)-[is related to]->(bushido)-[is related to]->(Hagakure)-[contains sayings attributed to]->(Nabeshima Naoshige)-[is representative figure of]->(Sengoku bushido)-[is from period]->(Muromachi-Azuchi (Sengoku period))

Case of KG-guided Synthesis: **Temporal Reasoning**

Question:
A Polish husband-and-wife sociological team introduced an English-language term for the meta-study of the scientific enterprise in a paper published in a year ending in 5. Exactly 11 years later, a quarterly journal dedicated to this field, but using a more versatile one-word term, was founded in their home country. Decades later, in a year divisible by 5, data was published indicating that a particular social science conferred a higher percentage of its doctorates on African-Americans than did a natural science, a field whose mathematical rigor is sometimes said to be a source of "envy" for the "softer" sciences. What is the duration, in years, between the founding of this one-word term quarterly journal and the publication of the Ph.D. distribution data?

Answer: 69 Year.

Reasoning Chain:
(the Ossowscy)-[introduced the term]-(Science of science)-[is also called]-(Logology)-[is the study of]-(science)-[is mismatched with]-(economics)-[is an example of]-(softer sciences)-[overuse]-(mathematics)-[is used in]-(social sciences)-[is a type of]-(psychology)-[has a higher proportion of African-American Ph.D.s than]-(physics)

Case of KG-guided Synthesis: **Causal Analysis**

Question:
In a novel first published in German in 1937, a fatal act of revenge by a Muslim protagonist against his Armenian rival is theorized to be a fictionalized account of the author's own youthful romantic frustrations. Correspondence from a town on the Amalfi Coast was instrumental in confirming the identity of this author, who wrote under a pseudonym. In an alternate timeline where the Central Powers triumphed in the global conflict of the 1910s, what was the resulting geopolitical stance of the nation where this author spent his final years?

Answer: Remained neutral through the entire war.

Reasoning Chain:
(Nachararyan)-[is rival of]-(Ali)-[murders]-(Nachararyan)-[is rival for love of]-(Nino)-[was basis for]-(Zhenia Flatt)-[was teenage love interest of]-(Nussimbaum)-[was receiving income as]-(Kurban Said)-[is identified as]-(Essad Bey)-[wrote letter in]-(Positano)-[is located in]-(Italy)-[is neutral in]-(Great War)