Temp-R1: A Unified Autonomous Agent for Complex Temporal KGQA via Reverse Curriculum Reinforcement Learning

Anonymous ACL submission

Abstract

Temporal Knowledge Graph Question Answering (TKGQA) is inherently challenging, as it requires sophisticated reasoning over dynamic facts with multi-hop dependencies and complex temporal constraints. Existing methods rely on fixed workflows and expensive closed-source APIs, limiting flexibility and scalability. We propose **Temp-R1**, the first autonomous end-to-end agent for TKGQA trained through reinforcement learning. To address cognitive overload in single-action reasoning, we expand the action space with specialized internal actions alongside external action. To prevent shortcut learning on simple questions, we introduce reverse curriculum learning that trains on difficult questions first, forcing the development of sophisticated reasoning before transferring to easier cases. Our 8B-parameter Temp-R1 achieves state-of-the-art performance on MULTITQ and TIMELINEKGQA, improving 19.8% over strong baselines on complex questions. Our work establishes a new paradigm for autonomous temporal reasoning agents.

1 Introduction

In the real world, knowledge is dynamic and constantly evolving. To prevent knowledge graphs from storing outdated facts, Temporal Knowledge Graphs have emerged, consisting of quadruples in the form of `<subject, predicate, object, timestamp>` (Jia et al., 2018a; Saxena et al., 2021). Consequently, Temporal Knowledge Graph Question Answering (TKGQA) requires reasoning over both entities and timestamps, making it significantly more challenging than conventional KGQA (Saxena et al., 2020). Complex Temporal queries involve multi-hop reasoning, multiple constraints (Su et al., 2024), and multi-granular time (Chen et al., 2023), which typically require combining external knowledge retrieval with the temporal reasoning capabilities of large language models (LLMs).

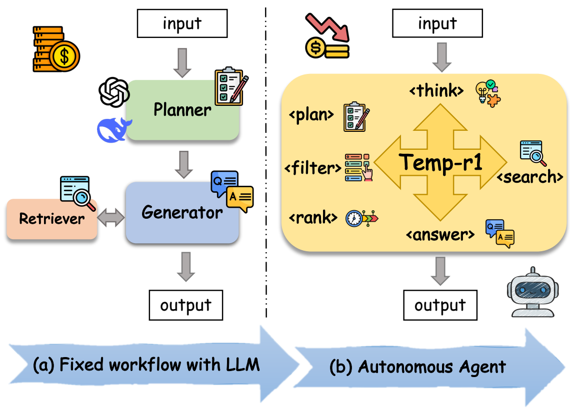

Figure 1: Paradigm shift in TKGQA systems. **Existing approaches** (a) suffer from fixed workflow and high API cost. **Temp-R1** (b) is an autonomous end-to-end agent trained for dynamic temporal reasoning.

Recent TKGQA methods have leveraged LLMs (Chen et al., 2024b; Gao et al., 2024; Jia et al., 2024). These approaches design fixed workflows comprising multiple carefully crafted prompts and components (decomposer, planner, generator), as illustrated in Figure 1(a). While achieving promising results, they heavily rely on the capabilities of closed-source models, incurring high API costs, as shown in Table 7 in Appendix D. Moreover, such manually designed fixed workflows constrain the model's divergent thinking, leading to rigid solution strategies. The field urgently needs a flexible paradigm that can autonomously perform reasoning and invoke TKG retrieval tools, rather than adhering to predetermined workflows.

The recent emergence of OpenAI o1 (OpenAI et al., 2024) and DeepSeek-R1 (Guo et al., 2025), along with growing research on autonomous LLM agents, offers new insights. Beyond their strong performance in mathematical reasoning and coding tasks, some methods combine reinforcement learning with search engine (Jin et al., 2025; Chen et al., 2025). However, applying these approaches to TKGQA exposes two critical challenges.

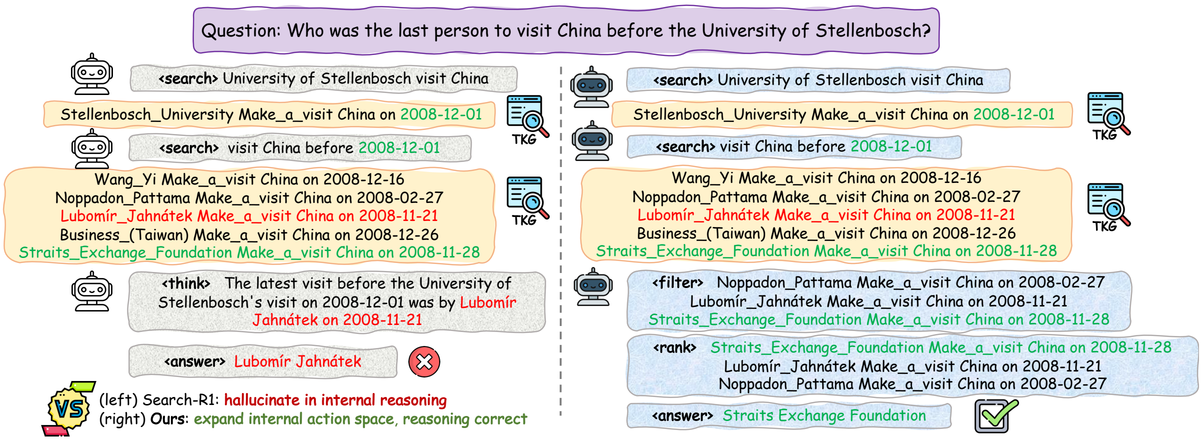

Figure 2: **Comparison of internal reasoning mechanisms: Temp-R1 vs. Search-R1.** By decoupling internal reasoning into explicit <filter> and <rank> actions (right), Temp-R1 maintains logical rigor and eliminates hallucinations in temporal sequencing. In contrast, the monolithic <think> block in Search-R1 (left) suffers from cognitive overload, leading to a failure in processing the retrieved temporal facts.

(1) Overloaded Internal Reasoning. Current search agents typically rely on a single internal reasoning tag, `<think>`, which carries excessive cognitive load. For complex temporal questions, the agent must simultaneously handle autonomous search, strategic planning, semantic filtering, and chronological ranking. Mixing these distinct cognitive demands within a single reasoning tag often leads to inadequate reinforcement learning and logical reasoning (as illustrated in Figure 2). **(2) The Shortcut Trap in Reinforcement Learning.** Temporal QA datasets (e.g., MULTITQ) usually have uneven difficulty distributions. Applying trainset directly causes the agent to overfit to simpler questions and stop exploring harder reasoning paths once achieving high rewards on easy samples. Agents gradually develop path dependency, making it difficult to activate complex tool combinations.

To address these challenges, we propose **Temp-R1**, a flexible autonomous agent that self-explores diverse solution strategies. By training an 8B parameter model that outperforms closed-source models, we significantly reduce inference costs, as shown in Figure 1(b). To enhance the internal reasoning capabilities, we expand the action space. Temp-R1 not only retains the external `<search>` action but also decouples three internal reasoning actions from the `<think>` token: `<plan>` enables initial problem analysis at rollout inception; `<filter>` applies semantic relations and temporal constraints to information; and `<rank>` orders filtered facts by timestamp. This compact yet expressive action design enables the agent to autonomously learn diverse reasoning trajectories through reinforcement learning (Kaelbling et al., 1996).

To overcome shortcut behavior in reinforcement learning, we adopt a counter-intuitive approach: after a brief supervised fine-tuning (SFT) warm-up, we employ **reverse curriculum learning**, forcing the model to survive in high-difficulty environments first, compelling it to acquire sophisticated tool-chain logic before transferring to simpler environments, achieving a "dimensional reduction" effect that enhances generalization.

Experimental results demonstrate that Temp-R1 achieves comprehensive superiority across three datasets including MULTITQ and TIMELINEKGQA. Notably, on the challenging `Multiple` question type, our model achieves a significant **19.8%** improvement over the strong baseline. Furthermore, our 8B parameter model establishes a new state-of-the-art overall performance, surpassing baselines built upon `GPT-4o`. Extensive ablation studies validate the effectiveness of reverse curriculum learning. In summary, our contributions are as follows:

- We propose a flexible autonomous agent for TKGQA, which decouples internal reasoning into an expanded action space, allowing for flexible and self-exploratory problem-solving.
- We introduce counter-intuitive reverse curriculum learning to mitigate the shortcut problem in reinforcement learning training.
- Extensive experiments show that our 8B model significantly outperforms powerful closed-source baselines, particularly on complex temporal reasoning tasks.

2 Related Work

2.1 Temporal Knowledge Graph QA

Early approaches to TKGQA can be broadly categorized into three methods. Embedding-based methods encode questions and TKG quadruples into low-dimensional vectors, ranking answers based on vector semantic similarity (Saxena et al., 2021; Mavromatis et al., 2022; Chen et al., 2023, 2022). Semantic parsing-based methods convert natural language questions into logical expressions to query TKGs, offering better interpretability (Jia et al., 2018b; Neelam et al., 2021; Chen et al., 2024a; Ding et al., 2022). Additionally, several studies have incorporated Graph Neural Networks to capture structural dependencies (Jia et al., 2021; Sharma et al., 2023; Liu et al., 2023b,a).

Recently, leveraging Large Language Models has become the mainstream paradigm. Some works focus on prompt engineering or fine-tuning: TimeR^4 (Qian et al., 2024) explicitly reveals implicit time constraints through question rewriting. TempAgent (Hu et al., 2025) adapts the ReAct paradigms to the temporal domain, designing a toolkit with 10 specific tools. RTQA (Gong et al., 2025) employs a decomposition-based approach to solve sub-problems recursively from the bottom up. MemoTime (Tan et al., 2025) utilizes closed-source model APIs for reasoning and stores solution paths of similar questions as memory. PoK (Qian et al., 2025) enhances TimeR^4 by generating plans for more comprehensive solution steps. However, these methods follow fixed workflows, which limits flexibility and results in high API costs.

2.2 Search Agent and Reinforcement Learning

Reinforcement Learning has evolved from PPO (Schulman et al., 2017) to more efficient direct optimization methods like DPO (Rafailov et al., 2023) and group-relative approaches like GRPO (Zheng et al., 2025). Unlike SFT, which requires token-level supervision, RL-based models optimize policies using verifiable reward signals.

Notable models like OpenAI-o1 (OpenAI et al., 2024) and DeepSeek-R1 (Guo et al., 2025) have demonstrated exceptional performance in mathematical logic and coding ability (Singh et al., 2025; Mai et al., 2025). In information retrieval, agents like Search-R1 (Jin et al., 2025; Song et al., 2025; Chen et al., 2025) apply these principles to autonomously search external knowledge.

3 Preliminary

TKG. A temporal knowledge graph $\mathcal{G} = \{\mathcal{E}, \mathcal{P}, \mathcal{T}, \mathcal{F}\}$ is a directed graph where vertices are a set of entities $\mathcal{E}$. The edges are a set of predicates $\mathcal{P}$ with timestamps $\mathcal{T}$. The quadruple set $\mathcal{F} = \{(s, p, o, t) \mid \mathcal{E} \times \mathcal{P} \times \mathcal{E} \times \mathcal{T}\}$ represents the temporal facts, where s and o are subject and object, respectively, and p is the predicate between s and o at timestamp t.

TKGQA. TKGQA is a task to infer the correct answer to a natural language question $q \in \mathcal{Q}$ based on relevant quadruples $f = (s, p, o, t)$ in the temporal knowledge graph $\mathcal{G}$. The answer can be an entity or a timestamp with varying granularities.

Agent-based MDP Formulation. Unlike conventional approaches, we formulate TKGQA as a Markov Decision Process (MDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$. The state space $\mathcal{S}$ consists of states $s_t = (q, h_t)$ where q is the original question and $h_t = [a_0, o_1, a_1, o_2, \ldots, a_{t-1}, o_t]$ represents the historical trajectory with actions a_i and observations o_i. The action space $\mathcal{A}$ encompasses both internal reasoning actions and external tool invocations. The transition function $\mathcal{P}(s_{t+1}|s_t, a_t)$ operates as follows: when a_t = <search>, the next observation is $o_{t+1} = \text{Retriever}(a_t)$; when $a_t \in$ {<plan>, <filter>, <rank>}, no external observation is generated ($o_{t+1} = \varnothing$); and when a_t = <answer>, the episode terminates with $s_{t+1} = s_{\text{terminal}}$. The reward function $\mathcal{R}(s_t, a_t)$ will be elaborated in Section 4.4.

4 Temp-R1

4.1 Overall Framework

In this section, we present **Temp-R1**, a unified framework designed to internalize autonomous temporal reasoning abilities into compact language models. As illustrated in Figure 3, our methodology unfolds as a progressive learning pipeline, encompassing: (1) establishing a **rollout loop** with expanded action space; (2) performing a **supervised cold start** to teach basic formatting and action sequences; (3) conducting **Group Relative Policy Optimization** for discovering autonomous reasoning strategies; and (4) deploying a **reverse curriculum learning strategy** to prevent shortcut traps and path dependency.

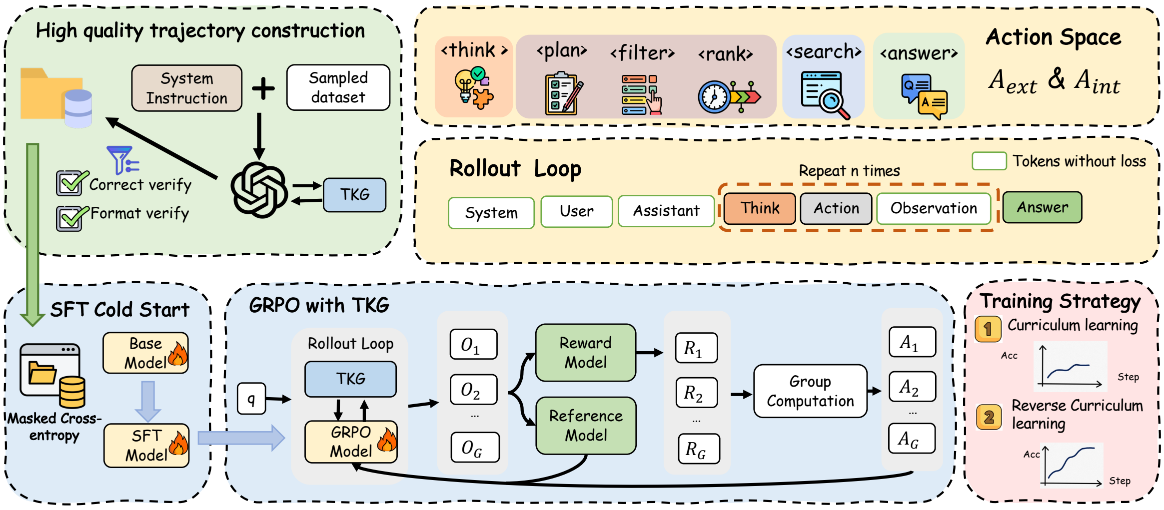

Figure 3: **Overall architecture of Temp-R1.** The rollout loop features an expanded internal and external action space ($\mathcal{A}_{internal}$ and $\mathcal{A}_{external}$) designed for structured temporal reasoning. The model is trained via a two-stage process: supervised cold start from high-quality trajectories, followed by GRPO-based reinforcement learning supported by reverse curriculum learning strategies.

4.2 Rollout Loop of Temp-R1

Based on the MDP defined in Sec. 3, the model explores the reasoning space through a structured rollout process. Unlike standard ReAct (Yao et al., 2023), our rollout interleaves mandatory strategic planning with autonomous information distillation.

Action Space Expansion. Temp-R1 decomposes reasoning into explicit actions rather than relying on a single thinking token. Internal actions include `<plan>`, `<filter>`, and `<rank>` for planning, filtering, and temporal organization, while the external action `<search>` retrieves knowledge. The final action `<answer>` terminates the rollout. This decomposition exposes intermediate reasoning states and facilitates long-horizon policy optimization.

Formally, the action space is defined as:

$$\mathcal{A} = \mathcal{A}_{\text{internal}} \cup \mathcal{A}_{\text{external}}, \qquad (1)$$

where internal actions are $\mathcal{A}_{\text{internal}} = \{$`<plan>`, `<filter>`, `<rank>`$\}$ and external actions are $\mathcal{A}_{\text{external}} = \{$`<search>`$\}$. The policy is modeled as $a_t \sim \pi_\theta(a_t|s_t)$, where s_t denotes the textual context up to step t and a_t is one of the internal or external actions. The state is updated after each action as $s_{t+1} = (q, h_t \cup \{a_t, o_{t+1}\})$, maintaining the complete interaction history for subsequent decision-making. The environment returns the corresponding observation such as retrieved information, which is then appended to the next input.

Prompt Template. Table 1 shows the system prompt of Temp-R1, which specifies the action space, planning requirements, and output format. The process stops when `<answer>` is produced or the maximum search turns $T_{\max}$ are reached.

4.3 Cold-Start: Learning Structured Format Alignment

Although the rollout mechanism enables an expanded action space, a pretrained base model without task-specific alignment struggles to produce valid structured tags and coherent reasoning in the early stages of training. To initialize a competent policy for reinforcement learning, we adopt a **Supervised Fine-Tuning (SFT)** phase as a cold start.

Data Construction. We construct a seed dataset $\mathcal{D}_{\text{sft}}$ consisting of (q, τ_{gold}) pairs, where τ_{gold} contains mandatory planning and basic reasoning primitives leading to correct answers. High quality trajectories are generated using `GPT-4o` and filtered to ensure structural correctness of reasoning tags and semantic correctness of the final answers.

Supervised Fine-Tuning. We train the model by minimizing the masked cross-entropy loss:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\frac{1}{T}\sum_{t=1}^{T} m_t \cdot \log \pi_\theta(x_t \mid x_{<t}), \qquad (2)$$

where x denotes the token sequence and $m_t \in \{0, 1\}$ is a selective loss mask. Specifically, the loss is applied only to agent-generated tokens (e.g.,

<plan>, <search>, <filter>, <answer>), while system prompts, user inputs, and retrieved observations are masked out. This selective token masking encourages the model to focus on structured reasoning and tool invocation, rather than memorizing retrieved content.

4.4 RL: Acquiring Autonomous Reasoning and Strategy Exploration

After SFT cold start, the model is further optimized with reinforcement learning to explore diverse reasoning strategies.

Group Relative Policy Optimization. Starting from the SFT-initialized policy $\pi_{\theta_{\text{old}}}$, we adopt Group Relative Policy Optimization (GRPO) to refine decision making. The clipped surrogate objective with KL regularization is defined as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\Big[\min\big(\rho_i(\theta)\hat{A}_i,\ \text{clip}(\rho_i(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_i\big) - \beta\,\mathbb{D}_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\Big], \tag{3}$$

where $\rho_i(\theta) = \frac{\pi_\theta(\tau_i|q)}{\pi_{\theta_{\text{old}}}(\tau_i|q)}$, ϵ is the clipping coefficient, and β controls the KL penalty against the reference policy π_{ref}. For each query q, we sample G trajectories $\{\tau_i\}_{i=1}^{G}$ from π_θ, each receiving a terminal binary reward $r_i \in \{0, 1\}$ based on answer correctness. The group relative advantage is computed as $\hat{A}_i = \frac{r_i - \text{mean}(\{r_k\})}{\text{std}(\{r_k\}) + \eta}$, where η is a small constant for numerical stability. This relative normalization encourages effective reasoning primitives and strategy exploration.

Reward Function. Temp-R1 is trained with a rule-based terminal reward that directly evaluates final answer correctness. The reward is defined as $R = 1$ if $a_{\text{pred}} = a_{\text{gold}}$, and $R = 0$ otherwise, where a_{pred} denotes the extracted final answer and a_{gold} is the ground truth.

4.5 Reverse Curriculum Learning Strategy

Traditional curriculum learning progresses from easy to hard ($\mathcal{D}_{\text{easy}} \rightarrow \mathcal{D}_{\text{medium}} \rightarrow \mathcal{D}_{\text{hard}}$), but this suffers from a shortcut trap in TKGQA tasks where models learn minimal patterns like <search> $\rightarrow$ <answer> that suffice for simple questions but fail to activate complex tool combinations needed for difficult temporal reasoning. Figure 8 empirically demonstrates that, without Reverse Curriculum Learning, training is dominated by path dependency and converges prematurely.

System Prompt

You are a question-answering assistant with a TKG.
Start with planning:
<plan>
- Question type:
- Time constraints:
- Sub-questions:
- Answer format:
</plan>
Then use tools as needed:
-<think>: Analytical reasoning. </think>
-<search> Search the TKG. </search>
-<information> Search results </information>.
-<filter> Filter facts by temporal constraints. </filter>
-<rank> Sort facts by date. </rank>
-<answer> Final answer </answer>
Important Constraints:
1. Every response MUST start with <plan>.
2. Answer format must match the specification.
3. Only use facts from retrieved information.
4. End with <answer> containing only the final answer.
Question: [Input Question]

Table 1: **System prompt** and action space template for Temp-R1. Different tags represent internal reasoning, external tools, and processing actions.

To overcome this, Temp-R1 employs a reverse curriculum strategy. We prioritize complex multi-hop queries $\mathcal{D}_{\text{multi}}$ to force the model to master sophisticated temporal reasoning and tool combinations first. Simpler queries $\mathcal{D}_{\text{single}}$ are introduced only after a warm-up threshold T_0:

$$\mathcal{D}_t = \begin{cases} \mathcal{D}_{\text{multi}}, & t \le T_0, \\ \mathcal{D}_{\text{multi}} \cup \mathcal{D}_{\text{single}}, & t > T_0. \end{cases} \tag{4}$$

This approach ensures that high-level reasoning capabilities are established early, which then generalize robustly to simpler tasks.

5 Experiments

5.1 Experimental Setups

Datasets. We train Temp-R1 on two datasets: MULTITQ and TIMELINEKGQA-CRON. We evaluate Temp-R1 on three benchmarks: MULTITQ, TIMELINEKGQA-CRON and TIMELINEKGQA-ICEWS-ACTOR. The TIMELINEKGQA-ICEWS-ACTOR dataset serves as an out-of-domain benchmark to evaluate the generalization ability of Temp-R1. Detailed statistics and category distributions are provided in Appendix A.

Baselines. We compare Temp-R1 against three types of baselines on MULTITQ: (1) **TKG Embedding-based methods**, including EmbedKGQA (Saxena et al., 2020), CronKGQA (Saxena et al., 2021), and MultiQA (Chen et al.,

Model	CronQuestion KG (In Domain)				ICEWS Actor (Out of Domain)			
	Overall	Simple	Medium	Complex	Overall	Simple	Medium	Complex
RAG Baseline	0.235	0.704	0.092	0.009	0.265	0.660	0.128	0.011
LLaMA2-7B	0.169	0.049	0.143	0.282	0.111	0.035	0.066	0.322
GPT-4o	0.206	0.069	0.130	0.376	0.113	0.051	0.035	0.353
RTQA	0.298	0.608	0.218	0.135	-	-	-	-
PoK	0.651	0.737	**0.539**	**0.683**	0.602	0.744	**0.456**	**0.578**
Temp-R1	**0.705**	**0.960**	0.486	0.672	**0.642**	**0.866**	0.388	0.595

Table 2: Performance comparison on TIMELINEKGQA across in-domain (CronQuestion KG) and out-of-domain (ICEWS Actor) scenarios. All results are evaluated using the **Hits@1** metric. The **best** and second best scores under each metric are highlighted in colors, respectively. Baseline results are from (Qian et al., 2025).

Model	Overall	Question Type		Answer Type	
		multiple	single	entity	time
TKG Embedding-based method					
EmbedKGQA	0.206	0.134	0.235	0.290	0.001
CronKGQA	0.279	0.134	0.337	0.328	0.156
MultiQA	0.293	0.159	0.347	0.349	0.157
Prompt-based LLM					
ARI	0.380	0.210	0.680	0.394	0.344
TempAgent	0.702	0.316	0.857	0.624	0.870
MemoTime	0.730	0.459	0.829	0.677	0.846
RTQA	0.765	0.424	0.902	0.692	0.942
FineTune-based LLM					
Search-R1	0.352	0.094	0.474	0.230	0.705
TimeR[4]	0.728	0.335	0.887	0.639	0.945
PoK	0.779	0.409	**0.929**	0.696	0.962
Temp-R1	**0.780**	**0.550**	0.888	**0.714**	**0.969**

Table 3: Performance comparison on the MULTITQ test set. Evaluation results are reported using the **Hits@1** metric across various question and answer categories. The **best** and second best scores under each metric are highlighted in colors, respectively.

Model	Overall	Question Type		Answer Type	
		multiple	single	entity	time
Temp-R1	**0.780**	**0.550**	**0.888**	**0.714**	**0.969**
w/o $\mathcal{A}_{int}$	0.620	0.388	0.729	0.563	0.783
w/o Reverse CL	0.556	0.143	0.750	0.447	0.868
w/o SFT	0.582	0.325	0.703	0.536	0.713

Table 4: **Ablation study** on MULTITQ using Hits@1. We evaluate the contribution of each key component by removing the internal actions ($\mathcal{A}_{int}$), the Reverse Curriculum Learning strategy, and the SFT cold-start.

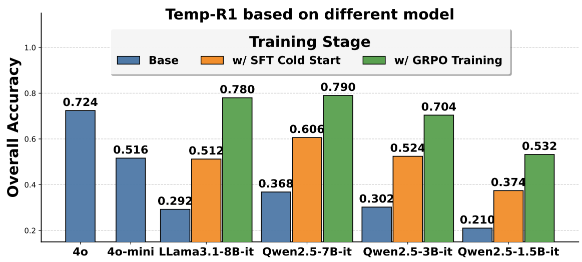

Figure 4: Performance comparison of Temp-R1 across different **backbone models** (`GPT-4o`, `GPT-4o-mini`, `LLaMA3.1-8B-it`, `Qwen2.5` series) and **training stages** (base model, w/SFT cold start, w/GRPO training).

2023); (2) **LLM with prompt engineering**, including ARI (Chen et al., 2024b), TempAgent (Hu et al., 2025), MemoTime (Tan et al., 2025), and RTQA (Gong et al., 2025); (3) **LLM with Fine-Tuning**, including Search-R1 (Jin et al., 2025), TimeR[4] (Qian et al., 2024), and PoK (Qian et al., 2025). For TIMELINEKGQA-CRON and TIMELINEKGQA-ICEWS-ACTOR, due to its complexity, existing embedding-based models are not directly applicable. We adopt five strong baselines. More baseline details are provided in Appendix B.

Implementation Details. We fine-tune the `Llama3.1-8B-Instruct` model, employing the same E5 retriever followed by (Jin et al., 2025). The SFT dataset consists of approximately 1,000 high-quality trajectories. The GRPO stage uses unlabeled QA pairs. Since the MULTITQ trainset is considerably large, we utilize only 9% of it for RL training. Detailed settings for both SFT and GRPO training config can be found in the Appendix C.

5.2 Main Results

The main experimental results on MULTITQ and TIMELINEKGQA are summarized in Table 3 and Table 2, respectively. Our proposed Temp-R1 outperforms all baseline methods across most metrics.

Overall Superiority and Model Efficiency. As shown in Table 3, Temp-R1 achieves the state-of-the-art performance with an overall score of 0.780 on MULTITQ. It is worth noting that while baselines rely on powerful closed-source LLMs (e.g., `GPT-4o-mini`, or `DeepSeek-V3`), our Temp-R1, based on an 8B open-source backbone, achieves

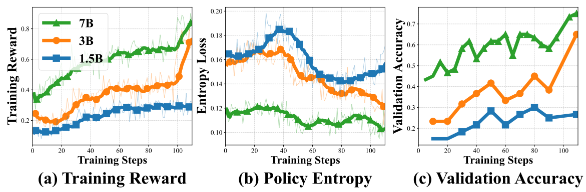

Figure 5: Training dynamics across different Qwen2.5 **backbone sizes** (7B, 3B, 1.5B). (a) Training Reward, (b) Policy Entropy, and (c) Val Accuracy

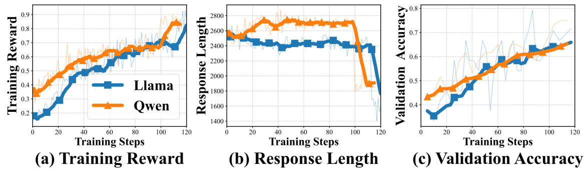

Figure 6: Training dynamics between LLaMA and Qwen **Architecture**. (a) Training Reward, (b) Response Length, and (c) Val Accuracy.

superior results. This demonstrates that a smaller model can surpass the performance of much larger, proprietary models that use fixed workflows.

Handling Complex Temporal Reasoning. A key strength of Temp-R1 is its ability to handle sophisticated temporal questions involving multi-hop reasoning and multiple constraints. **(1) Multiple Constraints:** In the `multiple` category of MULTITQ (Table 3), Temp-R1 achieves a score of 0.550, outperforming the state-of-the-art model by 9.1%. This improvement underscores its superior capacity for resolving intricate constraints that often challenge existing models. **(2) Multi-granular Precision:** The model demonstrates robust accuracy across diverse answer types, specifically reaching 0.969 on `time` category queries. This reflects high precision in pinpointing exact dates and intervals across various temporal granularities.

Generalization to Out-of-Domain Scenarios. Table 2 highlights the robustness of Temp-R1 when facing domain shifts. On the TIMELINE-ICEWS-ACTOR dataset, which serves as an Out-of-Domain testbed, Temp-R1 maintains its lead with an overall score of 0.610, surpassing *PoK* (0.602). While frozen models like `GPT-4o` struggle significantly in this specialized temporal domain (only 0.113), Temp-R1 exhibits remarkable stability.

5.3 Ablation Study

We evaluate the contribution of each component in **Temp-R1** via ablation experiments on MULTITQ, with results summarized in Table 4.

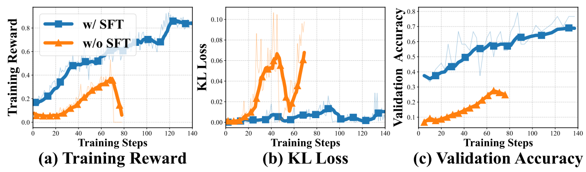

Figure 7: Training dynamics with/without **SFT Cold Start**. (a) Training Reward, (b) KL Loss, and (c) Val Accuracy. The blue line shows better stability and higher performance compared to the orange line.

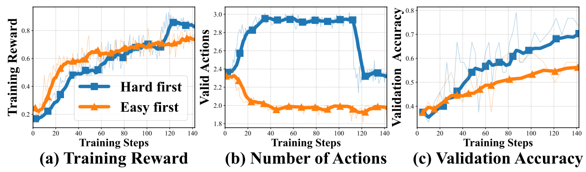

Figure 8: Training dynamics with/without **Reverse Curriculum Learning**. (a) Training Reward, (b) Number of Actions, (c) Val Accuracy. Without Reverse Curriculum Learning, all metrics show path dependency.

Effect of Internal Reasoning Actions. Removing $\mathcal{A}_{int}$ causes overall accuracy to drop from 0.780 to 0.620, with `Multiple` performance declining from 0.550 to 0.388. When only external actions are available, the model must perform all temporal reasoning implicitly in the <think> tag, which increases cognitive load and often leads to reasoning errors or even hallucinated conclusions.

Significance of SFT Cold Start. Removing SFT initialization reduces overall performance to 0.582, with time-type accuracy dropping from 0.969 to 0.713. Without the supervised fine-tuning initialization, the model starts reinforcement learning directly on challenging temporal tasks and fails to learn consistent reasoning formats. SFT provides essential prior knowledge of temporal patterns and formatting, serving as a bridge between static KG understanding and dynamic policy training.

Role of Reverse Curriculum Learning. Removing reverse curriculum learning reduces the overall score to 0.556. More critically, performance on hard queries drops sharply from 0.550 to 0.143, indicating that the agent overfits to simple questions and fails to solve complex reasoning tasks.

5.4 Impact of Backbone Model Selection

Model Scale: 7B, 3B vs. 1.5B. Figure 5 confirms that performance scales with parameter size. While all scales show steady improvement during training, the 7B model achieves peak accuracy of 0.790 versus 0.532 for the 1.5B variant. These results show that model capacity is critical for com-

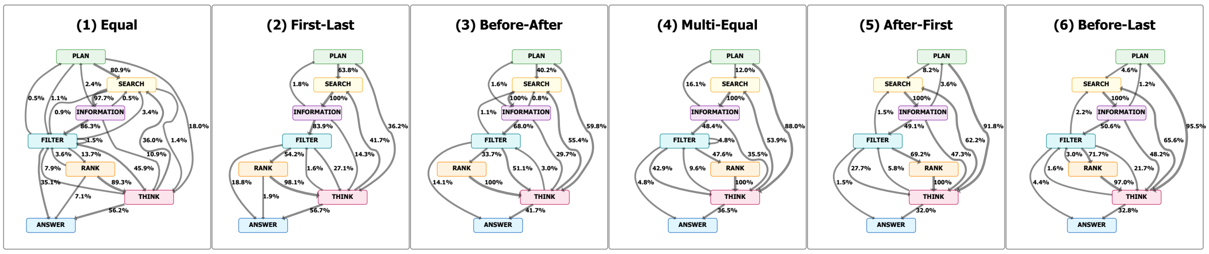

Figure 9: **Autonomous and flexible reasoning trajectories**: state transition diagrams across six question types.

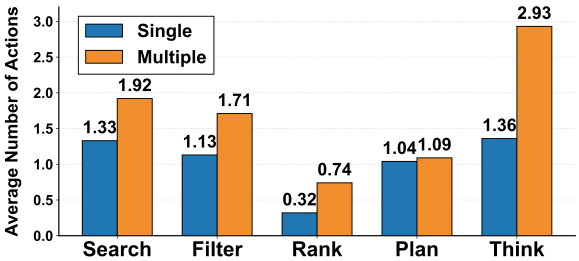

Figure 10: Comparison of the **average number of actions** for `Single` vs. `Multiple` tasks across internal action, external action and thinking.

plex reasoning, while validating that our training strategy remains effective across different scales.

Architecture Comparison: Llama vs. Qwen. Figure 6 demonstrates the broad adaptability of Temp-R1 across Llama and Qwen architectures. Both backbones exhibit stable training and comparable performance (0.790 for Qwen vs. 0.780 for Llama), with Qwen showing slightly longer response trajectories (Figure 6b). This confirms that Temp-R1 effectively enhances reasoning capabilities independent of underlying architecture.

5.5 Analysis of Training Strategy Choices

The Role of Cold Start: w/ SFT vs. w/o SFT. Figure 7 highlights SFT as a prerequisite for RL stability. Without SFT, the model suffers from early training collapse and uncontrollable KL loss spikes, leading to poor validation accuracy. SFT provides a high-quality initial policy, ensuring stable optimization and preventing the model from deviating into incoherent reasoning.

Effectiveness of GRPO Optimization: w/ GRPO vs. w/o GRPO. Figure 4 confirms the effectiveness of GRPO. Across all backbones, the "w/ GRPO Training" stage yields substantial accuracy gains over "Base" and "w/ SFT Cold Start" stages. This demonstrates that RL refinement is crucial for internalizing complex constraints and optimizing reasoning trajectories beyond imitation learning.

Curriculum Learning: Easy First vs. Hard First. Figure 8 highlights the advantages of reverse curriculum learning (*Hard First*). While the *Easy First* approach offers rapid initial reward growth, it often plateaus by relying on simplistic shortcuts. Conversely, the *Hard First* strategy promotes extensive exploration and higher action complexity (Figure 8b), avoiding local optima and achieving superior validation accuracy and generalization.

5.6 Reasoning Trajectory Analysis

Dynamics of Action State Transitions. As shown in Figure 9, Temp-R1 operates as a flexible agent rather than a rigid workflow. Transition probabilities vary notably across problem types, indicating adaptive trajectory planning. The model dynamically selects optimal action sequences according to query demands. Detailed case analysis is provided in Appendix E and Table 8–13.

Reasoning Complexity: Number of Actions. Figure 10 shows that the reasoning trajectory length scales with task difficulty. Compared to simple queries, complex problems trigger more internal actions, where <think> increase from 1.36 to 2.93 and <search> calls rise from 1.33 to 1.92. This adaptive behavior shows that the agent can autonomously devote more reasoning steps and tool calls when solving complex problems. Efficiency analysis is provided in Appendix D and Table 6– 7.

6 Conclusion

In this paper, we introduced **Temp-R1**, a novel autonomous agent framework that transforms TKGQA from a fixed-workflow paradigm into a flexible, self-exploratory reasoning process. Through expanded action space design and reverse curriculum learning, our 8B model achieves state-of-the-art performance, substantially outperforming `GPT-4o`-based methods. This work establish a new paradigm for building cost-effective, flexible autonomous reasoning agents.

Limitations

Due to computational constraints, our experiments are limited to models with parameter scales up to 8B. We did not conduct fine-tuning or reinforcement learning training on larger backbone models (such as 14B or beyond), which might further enhance temporal reasoning capability. Therefore, the scalability of Temp-R1 to higher-capacity models remains to be empirically verified.

Furthermore, while we innovatively integrated reverse curriculum learning into the TKGQA task and observed better performance gains than curriculum learning, these results do not necessarily imply that it is a universal solution for all reasoning tasks or data distributions. Our findings primarily demonstrate its efficacy within the TKGQA task, and its generalizability to broader, non-temporal domains requires further investigation.

Ethical considerations

In this paper, we investigate temporal knowledge graph question answering (TKGQA), focusing on complex reasoning over structured temporal data. Our method is developed and evaluated using publicly available and widely used datasets, including MULTITQ and TIMELINEKGQA. These datasets are constructed from open sources and do not contain any sensitive or personally identifiable information. Therefore, we believe that our work does not pose any ethical concerns.

References

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z. Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, and Weipeng Chen. 2025. Research: Learning to reason with search for llms via reinforcement learning. *CoRR*, abs/2503.19470.

Zhuo Chen, Zhao Zhang, Zixuan Li, Fei Wang, Yutao Zeng, Xiaolong Jin, and Yongjun Xu. 2024a. Self-improvement programming for temporal knowledge graph question answering. In *LREC/COLING*, pages 14579–14594. ELRA and ICCL.

Ziyang Chen, Dongfang Li, Xiang Zhao, Baotian Hu, and Min Zhang. 2024b. Temporal knowledge question answering via abstract reasoning induction. In *ACL (1)*, pages 4872–4889. Association for Computational Linguistics.

Ziyang Chen, Jinzhi Liao, and Xiang Zhao. 2023. Multi-granularity temporal question answering over knowledge graphs. In *ACL (1)*, pages 11378–11392. Association for Computational Linguistics.

Ziyang Chen, Xiang Zhao, Jinzhi Liao, Xinyi Li, and Evangelos Kanoulas. 2022. Temporal knowledge graph question answering via subgraph reasoning. *Knowl. Based Syst.*, 251:109134.

Wentao Ding, Hao Chen, Huayu Li, and Yuzhong Qu. 2022. Semantic framework based query generation for temporal question answering over knowledge graphs. In *EMNLP*, pages 1867–1877. Association for Computational Linguistics.

Yifu Gao, Linbo Qiao, Zhigang Kan, Zhihua Wen, Yongquan He, and Dongsheng Li. 2024. Two-stage generative question answering on temporal knowledge graph using large language models. In *ACL (Findings)*, pages 6719–6734. Association for Computational Linguistics.

Zhaoyan Gong, Juan Li, Zhiqiang Liu, Lei Liang, Huajun Chen, and Wen Zhang. 2025. RTQA : Recursive thinking for complex temporal knowledge graph question answering with large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9864–9881, Suzhou, China. Association for Computational Linguistics.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Qianyi Hu, Xinhui Tu, Cong Guo, and Shunping Zhang. 2025. Time-aware ReAct agent for temporal knowledge graph question answering. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6013–6024, Albuquerque, New Mexico. Association for Computational Linguistics.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018a. Tempquestions: A benchmark for temporal question answering. In *WWW (Companion Volume)*, pages 1057–1062. ACM.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018b. TEQUILA: temporal question answering over knowledge bases. In *CIKM*, pages 1807–1810. ACM.

Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. Faithful temporal question answering over heterogeneous sources. In *WWW*, pages 2052–2063. ACM.

Zhen Jia, Soumajit Pramanik, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Complex temporal question answering on knowledge graphs. In *CIKM*, pages 792–802. ACM.

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*.

Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore. 1996. Reinforcement learning: A survey. *J. Artif. Intell. Res.*, 4:237–285.

Yonghao Liu, Di Liang, Fang Fang, Sirui Wang, Wei Wu, and Rui Jiang. 2023a. Time-aware multiway adaptive fusion network for temporal knowledge graph question answering. In *ICASSP*, pages 1–5. IEEE.

Yonghao Liu, Di Liang, Mengyu Li, Fausto Giunchiglia, Ximing Li, Sirui Wang, Wei Wu, Lan Huang, Xiaoyue Feng, and Renchu Guan. 2023b. Local and global: Temporal question answering via information fusion. In *IJCAI*, pages 5141–5149. ijcai.org.

Xinji Mai, Haotian Xu, Xing W, Weinong Wang, Yingying Zhang, and Wenqiang Zhang. 2025. Agent RL scaling law: Agent RL with spontaneous code execution for mathematical problem solving. *CoRR*, abs/2505.07773.

Costas Mavromatis, Prasanna Lakkur Subramanyam, Vassilis N. Ioannidis, Adesoji Adeshina, Phillip Ryan Howard, Tetiana Grinberg, Nagib Hakim, and George Karypis. 2022. Tempoqr: Temporal question reasoning over knowledge graphs. In *AAAI*, pages 5825–5833. AAAI Press.

Sumit Neelam, Udit Sharma, Hima Karanam, Shajith Ikbal, Pavan Kapanipathi, Ibrahim Abdelaziz, Nandana Mihindukulasooriya, Young-Suk Lee, Santosh K. Srivastava, Cezar Pendus, Saswati Dana, Dinesh Garg, Achille Fokoue, G. P. Shrivatsa Bhargav, Dinesh Khandelwal, Srinivas Ravishankar, Sairam Gurajada, Maria Chang, Rosario Uceda-Sosa, and 6 others. 2021. SYGMA: system for generalizable modular question answering over knowledge bases. *CoRR*, abs/2109.13430.

OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, and 243 others. 2024. Openai o1 system card. *Preprint*, arXiv:2412.16720.

Xinying Qian, Ying Zhang, Yu Zhao, Baohang Zhou, Xuhui Sui, and Xiaojie Yuan. 2025. Plan of knowledge: Retrieval-augmented large language models for temporal knowledge graph question answering. *Preprint*, arXiv:2511.04072.

Xinying Qian, Ying Zhang, Yu Zhao, Baohang Zhou, Xuhui Sui, Li Zhang, and Kehui Song. 2024. Timer4 : Time-aware retrieval-augmented large language models for temporal knowledge graph question answering. In *EMNLP*, pages 6942–6952. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Apoorv Saxena, Soumen Chakrabarti, and Partha P. Talukdar. 2021. Question answering over temporal knowledge graphs. In *ACL/IJCNLP (1)*, pages 6663–6676. Association for Computational Linguistics.

Apoorv Saxena, Aditay Tripathi, and Partha P. Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *ACL*, pages 4498–4507. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Aditya Sharma, Apoorv Saxena, Chitrank Gupta, Seyed Mehran Kazemi, Partha P. Talukdar, and Soumen Chakrabarti. 2023. Twirgcn: Temporally weighted graph convolution for question answering over temporal knowledge graphs. In *EACL*, pages 2041–2052. Association for Computational Linguistics.

Joykirat Singh, Raghav Magazine, Yash Pandya, and Akshay Nambi. 2025. Agentic reasoning and tool integration for llms via reinforcement learning. *CoRR*, abs/2505.01441.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *CoRR*, abs/2503.05592.

Miao Su, Zixuan Li, Zhuo Chen, Long Bai, Xiaolong Jin, and Jiafeng Guo. 2024. Temporal knowledge graph question answering: A survey. *CoRR*, abs/2406.14191.

Xingyu Tan, Xiaoyang Wang, Qing Liu, Xiwei Xu, Xin Yuan, Liming Zhu, and Wenjie Zhang. 2025. Memotime: Memory-augmented temporal knowledge graph enhanced large language model reasoning. *Preprint*, arXiv:2510.13614.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *ICLR*. OpenReview.net.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. 2025. Group sequence policy optimization. *CoRR*, abs/2507.18071.

A Datasets & Corpora

We evaluate on three TKGQA benchmarks with complementary characteristics. MULTITQ offers large-scale QA pairs with diverse temporal granularities, while TIMELINEKGQA covers questions with varying complexity and time formats. Table 5 shows the detailed statistics.

MULTITQ. Built on ICEWS-Event with 461K temporal facts, MULTITQ contains 386K/58K/55K train/val/test questions. Questions are categorized as `Single` (73%, simple factual lookup or single-hop reasoning) or `Multiple` (27%, complex multi-hop reasoning with temporal ordering and constraints). The knowledge graph uses quadruples (subject, predicate, object, timestamp). Questions cover Before/After temporal sequencing, First/Last comparative reasoning, and both explicit/implicit temporal constraints across multiple granularities (day/month/year/entity). The imbalanced complexity distribution makes it ideal for studying curriculum learning and reward hacking in RL settings.

Timeline-ICEWS-Actor. Constructed from ICEWS political events with 108K facts, this dataset provides 54K/18K/18K train/val/test questions stratified into three balanced difficulty levels: `Simple` (33%, direct retrieval), `Medium` (30%, single-hop reasoning), and `Complex` (37%, multi-hop reasoning). The domain-specific focus on international relations and balanced difficulty distribution enable controlled evaluation of progressive learning strategies.

Timeline-CronQuestion. Derived from CronQuestions TKG with 329K facts and 25K/8K/8K train/val/test splits, this dataset emphasizes temporal intervals and duration reasoning. Questions span three complexity levels: `Simple` (29%), `Medium` (33%), and `Complex` (38%). Unlike the other datasets, it requires temporal arithmetic, interval calculations, and overlapping period reasoning, with answers including durations like “3 years”.

These datasets complement each other in scale, complexity distribution, and reasoning requirements, enabling comprehensive evaluation from simple retrieval to complex temporal inference.

B Baselines

We compare Temp-R1 against a comprehensive set of baselines covering different paradigms in temporal knowledge graph question answering (TKGQA). Specifically, we evaluate three categories of methods on MULTITQ, and five state-of-the-art baselines on TIMELINE-ICEWS-ACTOR and TIMELINE-CRONQUESTION. Our baseline selection represents the most recent and competitive approaches in the field, presenting significant challenges for evaluation.

(1) TKG Embedding-based Methods. These approaches learn low-dimensional temporal embeddings for entities and relations in TKGs, performing reasoning by computing similarity scores between question representations and candidate facts. Representative models include EmbedKGQA (Saxena et al., 2020), which extends static KG embedding techniques to question answering; CronKGQA (Saxena et al., 2021), which integrates temporal intervals into entity and relation embeddings; and MultiQA (Chen et al., 2023), which aggregates multi-granular temporal information for reasoning.

(2) LLM with Prompt Engineering. This category employs large language models as zero-shot or few-shot reasoners guided by carefully designed prompts. These methods represent particularly *strong baselines* as they leverage powerful closed-source model APIs, demonstrating capabilities fundamentally different from our approach. Representative examples include: ARI (Chen et al., 2024b) augments question understanding through adaptive reasoning instructions using `GPT-3.5-Turbo-0613`; TempAgent (Hu et al., 2025) adapts the *ReAct* framework with temporal reasoning toolkits powered by `GPT-4-Turbo`; MemoTime (Tan et al., 2025) retrieves past solution chains as external memory to enhance reasoning consistency, supporting multiple APIs including `GPT-4o-mini`, `GPT-4o`, `DeepSeek-V3`, and `DeepSeek-R1`. For fair comparison, we report results using `DeepSeek-V3` in our main experiments; RTQA (Gong et al., 2025) decomposes complex temporal questions into sequential sub-tasks solved recursively using `GPT-4o-mini` combined with `DeepSeek-V3`.

(3) LLM with Fine-Tuning. Instead of prompt engineering, these approaches fine-tune open-source language models to enhance temporal reasoning capabilities, though they still incorporate API calls or follow rigid predefined pipelines. Search-R1 (Jin et al., 2025) applies zero-shot reinforcement learning to teach the `Qwen2.5-7B` model the capability of deciding how and when to call ex-

Dataset	TKG	Question			
		Type	Train	Val	Test
MultiTQ	461,329	Single	283,482	41,735	38,864
		Multiple	103,305	16,244	15,720
		Total	386,787	57,979	54,584
Timeline-ICEWS-Actor	108,005	Simple	17,982	5,994	5,994
		Medium	15,990	5,330	5,330
		Complex	19,652	6,550	6,550
		Total	53,624	17,874	17,874
Timeline-CronQuestion	328,635	Simple	7,200	2,400	2,400
		Medium	8,252	2,751	2,751
		Complex	9,580	3,193	3,193
		Total	25,032	8,344	8,344

Table 5: Detailed statistics of the TKGQA datasets. Questions are categorized by difficulty or type, and the background TKG size is provided for each.

ternal search tools; TimeR^4 (Qian et al., 2024) explicitly exposes hidden temporal constraints through question rewriting and self-consistency filtering. Its pipeline includes a fine-tuned SentenceBERT retriever, `GPT-3.5-Turbo-0125` for rewriting, and a fine-tuned `LLaMA2-Chat-7B` generator; PoK (Qian et al., 2025) extends the rewriting strategy by introducing planning tokens that guide step-wise execution. It employs a fine-tuned `Qwen3-Embedding-0.7B` retriever, `GPT-4o` for planning, and a fine-tuned `LLaMA2-Chat-7B` generator.

Evaluation on Complex TIMELINEKGQA Datasets. For the more challenging TIMELINE-ICEWS-ACTOR and TIMELINE-CRONQUESTION datasets, which require sophisticated temporal interval reasoning and duration computation, existing embedding-based models are not directly applicable. Therefore, we adopt five strong neural and LLM-based baselines representing the current state of the art in timeline reasoning: RTQA and PoK. For the remaining three baselines, we report results directly from the PoK paper (Qian et al., 2025).

C Setup Details

This section outlines the configurations for the SFT and GRPO training stages.

Supervised Fine-Tuning (SFT). We fine-tune `LLaMA3.1-8B-Instruct` using LLaMA-Factory for 2 epochs with a learning rate of 2×10^{-5} and a batch size of 16. A key technical detail is the masking of $\langle$information$\rangle$ blocks to zero-out their loss, ensuring the model focuses on policy learning rather than redundant retrieval generation. Training employs `bf16` precision and AdamW optimizer with a 0.1 warmup ratio.

Group Relative Policy Optimization (GRPO). Post-SFT, reinforcement learning is conducted via a reverse curriculum on 3×A800 GPUs. We employ vLLM for rollouts with a temperature of 1.0 and a group size of 5. For GRPO, we set the clipping threshold $\epsilon = 0.2$ and KL penalty $\beta = 0.01$. The learning rate for the actor is 5×10^{-7}. FSDP and gradient clipping (5.0) are applied to ensure training stability.

D Efficiency Analysis

D.1 Training Cost Analysis

Table 6 compares the training cost of Temp-R1 and TimeR^4. Temp-R1 employs a unified agent trained primarily via reinforcement learning, requiring only 1,000 supervised instances for warm-up and approximately 35k rollout trajectories for optimization. In contrast, TimeR^4 relies on multi-stage supervised training with over 150k labeled instances and a separately trained retriever. Moreover, TimeR^4 depends on continuous external `GPT-3.5-turbo` API calls for question rewriting, introducing additional inference-time costs.

Overall, Temp-R1 achieves competitive performance with substantially fewer supervised signals and a simpler training pipeline, demonstrating improved training efficiency.

D.2 Inference-Time Cost Analysis

We further analyze inference-time efficiency by comparing Temp-R1 with representative prompt-based methods that rely on closed-source large lan-

	Temp-R1 (Ours)	TimeR[4]
Training Paradigm Model Components	RL (GRPO) + minimal SFT 1 (Unified agent)	Fully Supervised ≥3 (Retriever + LLM + Rewrite)
SFT Data Size SFT Epochs	1,000 2	∼75,357 (20% MULTITQ) 2
Main Optimization Data	∼35,100 rollout trajectories	∼150,714 supervised instances
Retriever Training Retriever Epochs Negative Sampling	– – –	Contrastive learning 10 1 in-batch + 3 hard negatives
External LLM Cost	∼$50 (one-time, SFT construction)	Continuous (GPT-3.5-turbo API call)

Table 6: **Comparison of Training Cost and Supervision Scale** between Temp-R1 and TimeR[4].

Model	Paradigm	LLM Access	API Calls	Inference Cost
TempAgent	Prompt-Engineering	Closed-source (GPT-4-turbo)	Multiple / query	High
RTQA	Prompt-Engineering	Closed-source (GPT-4o-mini + DeepSeek-V3)	Multiple / query	High
MemoTime	Prompt-Engineering	Closed-source (DeepSeek-V3)	Multiple / query	High
TimeR[4]	Fine-tuning + Prompting	Hybrid (GPT-3.5-turbo + LLaMA2)	Per query rewrite	Medium
PoK	Fine-tuning + Prompting	Hybrid (GPT-4o + LLaMA2)	Per query plan	Medium
Temp-R1	RL-trained Agent	Open-source (LLaMA3.1-8B)	None	Low

Table 7: **Inference-Time Cost Comparison** on MultiTQ. Prompt-based methods rely on closed-source LLM APIs during inference, while Temp-R1 performs fully local inference after training.

guage models. As summarized in Table 7, prompt-engineering approaches such as TempAgent (Hu et al., 2025), RTQA (Gong et al., 2025), and MemoTime (Tan et al., 2025) achieve strong performance but require multiple API calls to proprietary models during inference, resulting in continuous and non-negligible deployment costs.

In contrast, Temp-R1 is fully trained offline and performs inference using a single open-source model without any external API dependency. Once training is completed, no additional cost is incurred beyond standard model inference. This makes Temp-R1 more suitable for large-scale or long-term deployment scenarios where inference efficiency and cost stability are critical.

E Case Study

This section provides a detailed examination of the reasoning trajectories generated by **Temp-R1** across the six temporal question types in the MULTITQ dataset. For comparative temporal questions such as the "Before/After" type (Table 8), the trajectories demonstrate the model's ability to apply strict temporal filters to isolate events within a specified timeframe. In "Before-last" sequential reasoning scenarios (Table 9), the model successfully executes a multi-step logic by first identifying a reference event's date and then performing a secondary search to locate the most recent preceding occurrence. Similarly, for "First/Last" queries (Table 10), the model utilizes its ranking and chronological sorting modules to compare multiple retrieved candidates and identify the chronological extreme.

The model also demonstrates high precision in handling varied temporal constraints and anchoring events. For simple "Equal" factual questions (Table 11), the trajectory shows direct mapping between specific entities and their corresponding dates. In the more challenging "Multi-equal" multi-hop scenarios (Table 12), `Temp-R1` performs effective temporal bridging, using the time window of a pivot event to anchor the search for concurrent activities involving different entities. Finally, for "After-first" relative sequences (Table 13), the reasoning path highlights the model's ability to establish a historical anchor and chronologically evaluate subsequent events to find the immediate successor. Together, these trajectories confirm the robustness and structural clarity of **Temp-R1** in resolving multiple temporal granularities and multiple constraints.

Question: After May 2015, with which country did Oman formally sign an agreement?
Ground Truth: ['Japan' 'Malaysia' 'Iran']

Temp-R1:

<plan>
- Question type: Comparative temporal
- Time constraints: "After May 2015" (temporal scope)
- Sub-questions:
  1. Identify agreements signed by Oman after May 2015.
  2. Determine which countries were involved.
- Answer format: Country name

</plan>

<search> Oman agreement signed after May 2015 </search>

<information>
Doc 1(Title: Oman Sign_formal_agreement Malaysia on 2015-10-22)
Doc 2(Title: Oman Sign_formal_agreement Kuwait on 2015-02-19)
Doc 3(Title: Oman Sign_formal_agreement Iran on 2015-05-26)
Doc 4(Title: Malaysia Sign_formal_agreement Oman on 2015-10-22)
Doc 5(Title: Kuwait Sign_formal_agreement Oman on 2015-02-19)
Doc 6(Title: Oman Sign_formal_agreement Japan on 2015-06-23)
Doc 7(Title: Iran Sign_formal_agreement Oman on 2015-05-26)
Doc 8(Title: Japan Sign_formal_agreement Oman on 2015-06-23)
Doc 9(Title: Oman Sign_formal_agreement Iran on 2015-09-21)
Doc 10(Title: Iran Sign_formal_agreement Oman on 2015-09-21)
Doc 11(Title: United_Arab_Emirates Sign_formal_agreement Oman on 2005-05-24)
Doc 12(Title: Oman Sign_formal_agreement United_Arab_Emirates on 2005-05-24)
... (omitted 18 Docs)

</information>

<filter>
Doc 6(Title: Oman Sign_formal_agreement Japan on 2015-06-23)
Doc 9(Title: Oman Sign_formal_agreement Iran on 2015-09-21)

</filter>

<think> The filtered facts show two agreements signed by Oman after May 2015: with Japan on 2015-06-23 and Iran on 2015-09-21. </think>

<answer> Japan </answer>

Table 8: Case study using a before/after question.

Question: Who made the last request for Iraq before Médecins Sans Frontières?
Ground Truth: ['UN Security Council']

Temp-R1 Output:

<plan>
- Question type: Sequential temporal
- Time constraints: "last" (comparative - need most recent before another event)
- Sub-questions:
  1. When did Médecins Sans Frontières make the request for Iraq?
  2. Who made the request before that date?
  3. Identify the most recent request before the one by Médecins Sans Frontières.
- Answer format: Entity name

</plan>

<think> First, I need to find when the request for Iraq by Médecins Sans Frontières was made. </think>

<search> Médecins Sans Frontières request Iraq </search>

<information>
Doc 1(Title: Doctors_Without_Borders Make_an_appeal_or_request Iraq on 2014-06-18)
Doc 2(Title: Doctors_Without_Borders Make_an_appeal_or_request Niger on 2008-10-21)
Doc 3(Title: Reporters_Sans_Frontieres Make_an_appeal_or_request Interim_Government_(Iraq) on 2005-01-28)
Doc 4(Title: France Make_an_appeal_or_request Iraq on 2007-01-14)
Doc 5(Title: Doctors_Without_Borders Make_an_appeal_or_request Other_Authorities_/_Officials_(Chad) on 2014-04-03)
Doc 6(Title: International_Government_Organizations Make_an_appeal_or_request Iraq on 2007-02-18)
Doc 7(Title: France Make_an_appeal_or_request Iraq on 2006-11-06)
Doc 8(Title: France Make_an_appeal_or_request Iraq on 2006-01-29)
Doc 9(Title: Doctors_Without_Borders Make_an_appeal_or_request Other_Authorities_/_Officials_(Chad) on 2014-04-02)
Doc 10(Title: France Make_an_appeal_or_request Iraq on 2014-08-13)
... (omitted 20 Docs)

</information>

<think> The request for Iraq by the "Doctors_Without_Borders" was on 2014-06-18. Now, I need to find who made the most recent request before this date. </think>

<search> request to Iraq before 2014-06-18 </search>

<information>
Doc 1(Title: UN_Security_Council Make_an_appeal_or_request Iraq on 2014-06-06)
Doc 2(Title: Barack_Obama Make_an_appeal_or_request Iraq on 2014-06-24)
Doc 3(Title: Barack_Obama Make_an_appeal_or_request Iraq on 2014-06-23)
Doc 4(Title: Barack_Obama Make_an_appeal_or_request Iraq on 2014-06-19)
Doc 5(Title: Barack_Obama Make_an_appeal_or_request Iraq on 2014-06-21)
Doc 6(Title: Nuri_al-Maliki Make_an_appeal_or_request Iraq on 2014-06-25)
Doc 7(Title: UN_Security_Council Make_an_appeal_or_request Iraq on 2014-05-01)
Doc 8(Title: Iraq Make_an_appeal_or_request Barack_Obama on 2014-06-12)
Doc 9(Title: Barack_Obama Make_an_appeal_or_request Iraq on 2014-07-02)
Doc 10(Title: Barack_Obama Demand Iraq on 2014-06-19)
... (omitted 20 Docs)

</information>

<filter>
Doc 1 (Title: UN_Security_Council Make_an_appeal_or_request Iraq on 2014-06-06)

</filter>

<think> The most recent request before the one by the "Doctors_Without_Borders" on 2014-06-18 was by the UN Security Council on 2014-06-06. </think>

<answer> UN Security Council </answer>

Table 9: Case study using a before-last question.

Question: In which year did the Chancellor of the Exchequer visit China for the first time?

Ground Truth: ['2005']

Temp-R1 Output:

<plan>
- Question type: Multi-hop
- Time constraints: “first time” (comparative - need earliest date)
- Sub-questions:
  1. When did the Chancellor of the Exchequer visit China?
  2. Identify the earliest such visit.
- Answer format: yyyy

</plan>

<think> First, I need to find visits to China by the Chancellor of the Exchequer and identify the earliest one. </think>

<search> Chancellor of the Exchequer visit China </search>

<information>
Doc 1(Title: Chancellor_of_the_Exchequer Make_a_visit China on 2005-02-21)
Doc 2(Title: Chancellor_of_the_Exchequer Make_a_visit China on 2015-09-22)
Doc 3(Title: China Host_a_visit Chancellor_of_the_Exchequer on 2005-02-21)
Doc 4(Title: China Host_a_visit Chancellor_of_the_Exchequer on 2015-09-22)
Doc 5(Title: Chancellor_of_the_Exchequer Express_intent_to_meet_or_negotiate China on 2012-01-14)
Doc 6(Title: Chancellor_of_the_Exchequer Express_intent_to_meet_or_negotiate China on 2012-01-13)
Doc 7(Title: Chancellor_of_the_Exchequer Make_statement China on 2005-02-21)
Doc 8(Title: Wen_Jiabao Express_intent_to_meet_or_negotiate Chancellor_of_the_Exchequer on 2009-01-22)
Doc 9(Title: Wen_Jiabao Consult Chancellor_of_the_Exchequer on 2005-02-21)
Doc 10(Title: Chancellor_of_the_Exchequer Consult Wen_Jiabao on 2005-02-21)
... (omitted 20 Docs)

</information>

<filter>
Doc 1 (Chancellor of the Exchequer Make a visit China on 2005-02-21)
Doc 2 (Chancellor of the Exchequer Make a visit China on 2015-09-22)

</filter>

<rank>
Doc 1 (Chancellor of the Exchequer Make a visit China on 2005-02-21)
Doc 2 (Chancellor of the Exchequer Make a visit China on 2015-09-22)

</rank>

<think> The earliest visit of the Chancellor of the Exchequer to China was on 2005-02-21. </think>

<answer> 2005 </answer>

Table 10: Case study using a first/last question.

Question: When did Mswati III denounce Swaziland?
Ground Truth: ['2011-03-19']

Temp-R1 Output:

<plan>
- Question type: Simple factual
- Time constraints: None explicitly stated
- Sub-questions: None needed
- Answer format: yyyy-mm-dd

</plan>

<search> Mswati III denounce Swaziland </search>

<information>
Doc 1(Title: Mswati_III Criticize_or_denounce Swaziland on 2011-03-19)
Doc 2(Title: Swaziland Criticize_or_denounce Mswati_III on 2011-04-12)
Doc 3(Title: Swaziland_Solidarity_Network Criticize_or_denounce Mswati_III on 2011-03-05)
Doc 4(Title: Mswati_III Reject Legislature_(Swaziland) on 2012-11-15)
Doc 5(Title: Mswati_III Make_statement Swaziland on 2006-02-10)
Doc 6(Title: Mswati_III Make_statement Swaziland on 2007-09-29)
Doc 7(Title: Mswati_III Make_statement Swaziland on 2005-08-16)
Doc 8(Title: Mswati_III Make_statement Swaziland on 2008-09-05)
Doc 9(Title: Mswati_III Make_statement Swaziland on 2011-09-14)
Doc 10(Title: Mswati_III Make_statement Swaziland on 2012-07-25)
... (omitted 20 Docs)

</information>

<filter>
Doc 1: Mswati_III Criticize_or_denounce Swaziland on 2011-03-19

</filter>

<answer> 2011-03-19 </answer>

Table 11: Case study using a equal question.

Question: Who was Kuwait negotiating with in the same month as Christine Lagarde?
Ground Truth: ['UN Security Council']

Temp-R1 Output:

<plan>
- Question type: Multi-hop
- Time constraints: "in the same month as Christine Lagarde" (temporal scope)
- Sub-questions:
  1. When did Christine Lagarde have relevant negotiations or event with Kuwait?
  2. Who was Kuwait negotiating with in that same month?
- Answer format: Entity name or concise phrase

</plan>

<think> First, I need to find out when Christine Lagarde had relevant negotiations or event with Kuwait. </think>

<search> Christine Lagarde negotiations Kuwait </search>

<information>
Doc 1(Title: Christine_Lagarde Engage_in_negotiation Kuwait on 2009-06-21)
Doc 2(Title: Kuwait Engage_in_negotiation Christine_Lagarde on 2009-06-21)
Doc 3(Title: Christine_Lagarde Consider_policy_option Kuwait on 2015-11-11)
Doc 4(Title: Christine_Lagarde Engage_in_negotiation Abdel_Fattah_Al-Sisi on 2014-06-07)
Doc 5(Title: Christine_Lagarde Make_statement Kuwait on 2009-06-22)
Doc 6(Title: Abdel_Fattah_Al-Sisi Engage_in_negotiation Christine_Lagarde on 2014-06-07)
Doc 7(Title: Christine_Lagarde Express_intent_to_engage_in_diplomatic_cooperation_(such_as_policy_support) UAE on 2006-11-27)
Doc 8(Title: Christine_Lagarde Express_intent_to_meet_or_negotiate Middle_East on 2014-05-06)
Doc 9(Title: Christine_Lagarde Engage_in_negotiation Yannis_Stournaras on 2014-06-05)
Doc 10(Title: Christine_Lagarde Express_intent_to_meet_or_negotiate Central_Bank_(Jordan) on 2014-05-06)
... (omitted 20 Docs)

</information>

<think> The relevant negotiation or event with Christine Lagarde and Kuwait occurred on 2009-06-21. Now I need to find out who Kuwait was negotiating with in June 2009. </think>

<search> Kuwait negotiation June 2009 </search>

<information>
Doc 1(Title: Kuwait Engage_in_negotiation Iraq on 2009-07-27)
Doc 2(Title: Kuwait Engage_in_negotiation Oman on 2008-06-10)
Doc 3(Title: UN_Security_Council Engage_in_negotiation Kuwait on 2009-06-03)
Doc 4(Title: UAE Engage_in_negotiation Kuwait on 2008-06-10)
Doc 5(Title: Oman Engage_in_negotiation Kuwait on 2008-06-10)
Doc 6(Title: Iraq Engage_in_negotiation Kuwait on 2009-07-27)
Doc 7(Title: Kuwait Engage_in_negotiation UN_Security_Council on 2009-06-03)
Doc 8(Title: Kuwait Engage_in_negotiation UAE on 2008-06-10)
Doc 9(Title: Kuwait Engage_in_negotiation Christine_Lagarde on 2009-06-21)
Doc 10(Title: Christine_Lagarde Engage_in_negotiation Kuwait on 2009-06-21)
... (omitted 20 Docs)

</information>

<filter>
Doc 9(Title: Kuwait Engage_in_negotiation Christine_Lagarde on 2009-06-21)
Doc 7(Title: Kuwait Engage_in_negotiation UN_Security_Council on 2009-06-03)

</filter>

<think> The filtered facts show that in June 2009, besides Christine Lagarde, the UN Security Council was involved in negotiations with Kuwait. </think>

<answer> UN Security Council </answer>

Table 12: Case study using a multi-equal question.

Question: After Ethiopia, against which country did South Sudan first use conventional military force?

Ground Truth: ['Sudan']

Temp-R1 Output:

<plan>
- Question type: Sequential temporal
- Time constraints: "first use after Ethiopia" (sequential - need earliest after a specific event)
- Sub-questions:
  1. When did South Sudan use conventional military force against Ethiopia?
  2. What other countries has South Sudan used conventional military force against, and when?
  3. Among these events, which occurred first after the one against Ethiopia?
- Answer format: Country name

</plan>

<think> First, I need to find when South Sudan used conventional military force against Ethiopia. </think>

<search> South Sudan use conventional military force Ethiopia </search>

<information>
Doc 1(Title: South_Sudan Use_conventional_military_force Ethiopia on 2012-01-06)
Doc 2(Title: South_Sudan Use_conventional_military_force Combatant_(Lord's_Resistance_Army) on 2008-06-16)
Doc 3(Title: South_Sudan Use_conventional_military_force Combatant_(Lord's_Resistance_Army) on 2008-06-17)
Doc 4(Title: Sudan_People's_Liberation_Army Use_conventional_military_force South_Sudan on 2011-02-14)
Doc 5(Title: Sudan_People's_Liberation_Army Use_conventional_military_force South_Sudan on 2014-01-30)
Doc 6(Title: Sudan_People's_Liberation_Army Use_conventional_military_force Military_(South_Sudan) on 2012-04-14)
Doc 7(Title: Sudan_People's_Liberation_Army Use_conventional_military_force South_Sudan on 2009-06-14)
Doc 8(Title: Sudan_People's_Liberation_Army Use_conventional_military_force South_Sudan on 2014-02-02)
Doc 9(Title: Sudan_People's_Liberation_Army Use_conventional_military_force Military_(South_Sudan) on 2013-01-14)
Doc 10(Title: Sudan_People's_Liberation_Army Use_conventional_military_force Military_(South_Sudan) on 2013-01-29)
... (omitted 20 Docs)

</information>

<think> The date when South Sudan used conventional military force against Ethiopia was 2012-01-06. Now, I need to find which country South Sudan used conventional military force against after this date and identify the earliest occurrence. </think>

<search> South Sudan use conventional military force after 2012-01-06 </search>

<information>
Doc 1(Title: Military_(South_Sudan) Use_conventional_military_force South_Sudan on 2012-01-04)
Doc 2(Title: Sudan_People's_Liberation_Army Use_conventional_military_force Military_(South_Sudan) on 2012-04-14)
Doc 3(Title: Sudan_People's_Liberation_Army Use_conventional_military_force Military_(South_Sudan) on 2012-04-23)
Doc 4(Title: Military_(South_Sudan) Use_conventional_military_force South_Sudan on 2012-08-24)
Doc 5(Title: Military_(South_Sudan) Use_conventional_military_force South_Sudan on 2012-08-25)
Doc 6(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-26)
Doc 7(Title: Government_(Sudan) Use_conventional_military_force South_Sudan on 2012-05-23)
Doc 8(Title: Military_(South_Sudan) Use_conventional_military_force Indigenous_People_(South_Sudan) on 2012-05-24)
Doc 9(Title: South_Sudan Use_conventional_military_force Ethiopia on 2012-01-06)
Doc 10(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-29)
... (omitted 20 Docs)

</information>

<filter>
Doc 10(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-29)
Doc 6(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-26)

</filter>

<rank>
Doc 6(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-26)
Doc 10(Title: South_Sudan Use_conventional_military_force Sudan on 2012-02-29)

</rank>

<think> The earliest use of South Sudan's conventional military force after the one against Ethiopia on 2012-01-06 is against Sudan on 2012-02-26. </think>

<answer> Sudan </answer>

Table 13: Case study using a after-first question.