TimelineReasoner: Advancing Timeline Summarization with Large Reasoning Models

Liancheng Zhang
Renmin University of China
Beijing, China
zhangliancheng227@ruc.edu.cn

Xiaoxi Li
Renmin University of China
Beijing, China
xiaoxi_li@ruc.edu.cn

Zhicheng Dou*
Renmin University of China
Beijing, China
dou@ruc.edu.cn

Abstract

The proliferation of online news poses a challenge to extracting structured timelines from unstructured content. While recent studies have shown that Large Language Models (LLMs) can assist Timeline Summarization (TLS), these approaches primarily treat models as passive generators. The emergence of Large Reasoning Models (LRMs) presents an opportunity to reason over events actively, enabling iterative evidence acquisition, the detection of missing events, and the validation of temporal consistency. To leverage the reasoning capabilities of LRMs, we propose **TimelineReasoner**, a novel framework that shifts TLS from static generation to an active, reasoning-driven process. TimelineReasoner adopts a two-stage framework: Global Cognition, which tracks events at a macroscopic level and continuously updates a global event memory, and Detail Exploration, which identifies informational gaps and refines the timeline via targeted document retrieval. To support this, TimelineReasoner incorporates several specialized mechanisms, including an Event Scraper for retrieving temporal event descriptions, a Timeline Updater for refining the timeline, and a Supervisor for detecting gaps in the timeline and guiding retrieval. Experimental results on open-domain TLS datasets demonstrate that TimelineReasoner significantly outperforms existing LLM-based TLS methods in terms of timeline accuracy, coverage, and coherence. On closed-domain TLS datasets, our method performs on par with or exceeds state-of-the-art approaches. This work not only pushes the boundaries of TLS but also highlights the broader potential of LRM-based reasoning frameworks for timeline summarization.

CCS Concepts

• **Information systems** → *Information systems applications.*

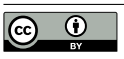
Keywords

Timeline Summarization, Deep Research, Large Reasoning Model

ACM Reference Format:

Liancheng Zhang, Xiaoxi Li, and Zhicheng Dou. 2026. TimelineReasoner: Advancing Timeline Summarization with Large Reasoning Models. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '26), July 20–24, 2026, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3805712.3809710

*Corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License.
SIGIR '26, Melbourne, VIC, Australia
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2599-9/2026/07
https://doi.org/10.1145/3805712.3809710

1 Introduction

In today's digital era, the rapid proliferation of news information presents a significant challenge: extracting coherent event narratives from an overwhelming volume of textual sources. Every minute, countless news articles are published, making it difficult for readers to track the evolution of complex events. *Timeline Summarization (TLS)* [1, 4, 11, 39] addresses this problem by distilling chronological event descriptions from large corpora, producing structured summaries that capture key developments over time.

Recent advances in Large Language Models (LLMs) [2, 17, 30] have introduced new opportunities for TLS, leveraging their superior comprehension and generation capabilities to produce high-quality summaries. However, existing LLM-based approaches [15, 35] primarily treat these models as passive generators, which means feeding them raw articles and directly generating timeline outputs. As illustrated in Figure 1 (a), such methods typically construct an event graph and then generate the final timeline from it. While effective at extracting local details, these methods lack a global mechanism for dynamically acquiring and integrating event information across the entire timeline, making it difficult to ensure coherent coverage and temporal consistency. Retrieval and summarization are performed in isolation, without validating the logical continuity of events. As a result, timelines often suffer from fragmentation, redundancy, and temporal inconsistency.

Meanwhile, Large Reasoning Models (LRMs) [8, 16, 33] have demonstrated remarkable problem-solving abilities in domains like mathematics, coding, and scientific research. Their capability for structured reasoning, and tool-augmented workflows makes them well-suited for tasks requiring multi-step analysis, which are precisely the demands of TLS. In particular, applying LRMs to TLS is promising, as their deep, multi-step reasoning capabilities can support dynamic evidence acquisition, missing-event discovery, and temporal consistency verification across long timelines. However, the field still lacks a principled framework that models timeline construction as a hierarchical cognitive process. Such a framework should move beyond passively generating timelines, and instead dynamically construct comprehensive and temporally accurate timelines through iterative reasoning and targeted information retrieval.

To address this, we propose **TimelineReasoner**, a reasoning-driven framework that actively acquires, integrates, and refines event information for TLS. TimelineReasoner leverages the capabilities of LRMs to dynamically construct timelines through iterative reasoning and targeted information retrieval. Our framework adopts a two-stage design—**Global Cognition** and **Detail Exploration**—supported by specialized mechanisms.

In the Global Cognition stage, TimelineReasoner aims to establish an initial, coarse-grained understanding of the overall event

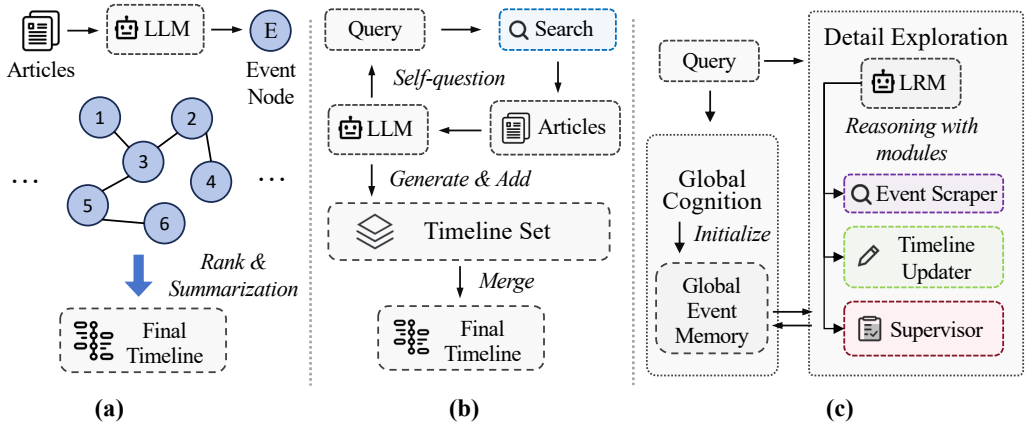


Figure 1: Comparison of methods using LLMs for TLS. (a) A method that uses LLMs to generate event nodes and constructs an event graph for timeline derivation via cluster ranking. (b) A framework that employs iterative self-questioning to link events, merging generated sub-timelines into a final output. (c) TimelineReasoner (Ours): An active, reasoning-driven framework using LRMs. It maintains a dynamic global event memory during Global Cognition and employs an agentic loop for Detail Exploration to iteratively refine the timeline.

landscape, identifying the major events and their approximate temporal structure to form a global view of the evolving news narrative. This stage is initialized through a broad, one-pass retrieval process that gathers representative news articles, from which high-level event information is extracted by the **Event Scraper**. The resulting global understanding serves as a foundation for subsequent fine-grained refinement, and is later updated using newly discovered evidence from the Detail Exploration stage.

In the Detail Exploration stage, TimelineReasoner focuses on identifying missing or incomplete events and acquiring additional relevant information to fill these gaps, ensuring both detail completeness and temporal consistency. It performs a search-interleaved reasoning process, analyzing the global event memory and the evolving timeline memory to identify informational gaps and generate targeted sub-queries to the Event Scraper for fine-grained event retrieval, and produce sub-timelines that are integrated into the timeline memory via the **Timeline Updater**. A **Supervisor** module further monitors coverage gaps, temporal inconsistencies, and informational deficiencies, guiding the LRM with structured search plans to iteratively refine the timeline until it achieves high completeness, coherence, and factual accuracy.

This reasoning-centric architecture enables TimelineReasoner to go beyond static summarization, actively bridging missing temporal links, resolving inconsistencies, and delivering structured, high-fidelity timelines from unstructured multi-document news corpora.

In summary, our main contributions are as follows:

(1) We introduce a coarse-to-fine reasoning paradigm for timeline construction that decouples *global event comprehension* from *local detail verification*. This paradigm formulates timeline summarization as an iterative reasoning process that progressively refines the timeline through targeted evidence acquisition, addressing the fundamental challenges of temporal inconsistency, incomplete coverage, and fragmented narratives in existing TLS methods.

(2) Building on this paradigm, we propose **TimelineReasoner**, a reasoning-driven framework that instantiates the coarse-to-fine process via modular components for event acquisition, timeline refinement, and consistency maintenance. The framework enables active interaction between reasoning and retrieval, allowing the timeline to be incrementally improved rather than passively generated in a single step.

(3) We design a *dynamic timeline memory updating mechanism* that continuously integrates newly retrieved evidence into an evolving timeline while preserving temporal and factual consistency. This mechanism allows TimelineReasoner to revise and enrich previous timeline memory as new information becomes available.

(4) We introduce an **agentic supervision mechanism** for TLS, which performs meta-level analysis over the evolving timeline to identify coverage gaps, temporal inconsistencies, and informational deficiencies. This supervision mechanism generates structured search plans to guide TimelineReasoner toward more targeted retrieval and reasoning, thereby improving the completeness and coherence of the resulting timelines.

2 Related Work

2.1 Timeline Summarization

Timeline Summarization (TLS) aims to construct a coherent, chronologically ordered account of event evolution from a collection of documents [1, 4, 11, 39]. The task requires modeling temporal dependencies, identifying salient events, and reducing redundancy across heterogeneous sources. Depending on whether the document collection is fixed or dynamically retrieved, TLS is commonly categorized into closed-domain and open-domain settings [35].

Closed-domain TLS is often framed as a specialized form of multi-document summarization, where timelines are generated from a static document set [7, 20, 29, 41]. Prior work has mainly focused on detecting salient dates via temporal features and graphical models [5, 11, 31], or identifying major events through graph-based

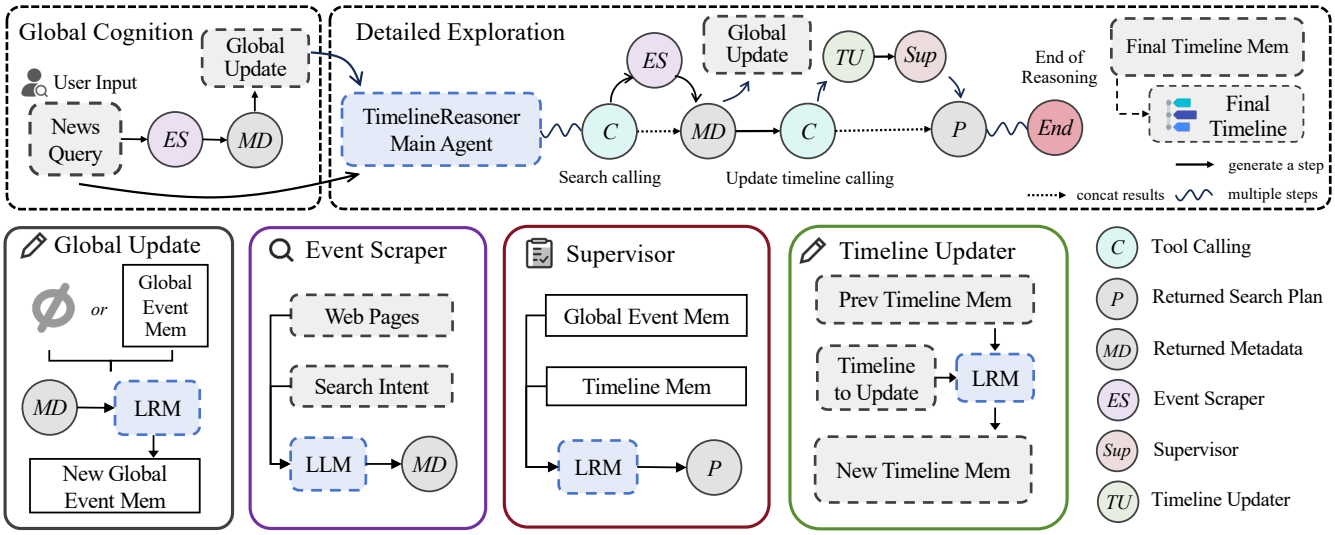


Figure 2: Framework of TimelineReasoner. It contains two stages: Global Cognition searches for the news query, uses an LRM to generate global event memory, and dynamically updates it in the later process. Detailed Exploration takes a main reasoning model as the core agent, which is equipped with several specialized mechanisms, Event Scraper, Timeline Updater, and Supervisor, to dynamically refine the timeline memory to obtain the final timeline.

and clustering techniques [9, 11, 21, 38, 39]. While effective in constrained scenarios, these approaches typically rely on heuristic temporal signals and struggle to capture long-range event dependencies and global narrative coherence.

More recently, LLMs have influenced TLS research, particularly in open-domain settings where document collections are dynamically retrieved. Hu et al. [15] leveraged LLMs to generate event-centric summaries and cluster them into timelines, while Wu et al. [35] proposed an iterative self-questioning framework to guide retrieval and refinement. Although these methods improve flexibility, they largely treat LLMs as generative or prompt-driven components, without explicitly modeling timeline construction as a structured reasoning process. In contrast, our work explores how LRMs can support TLS through hierarchical reasoning and search, enabling dynamic evidence acquisition, iterative refinement, and principled temporal consistency verification.

2.2 Large Reasoning Model (LRM)

LRMs are distinguished from conventional LLMs by their emphasis on improving test-time performance through extended reasoning processes, rather than solely scaling parameters or data [3, 13, 36, 40]. Unlike traditional LLMs that operate primarily as direct text generators, LRMs incorporate structured reasoning workflows that iteratively refine intermediate representations. Representative models such as OpenAI-o1 [16], Qwen-QwQ [33], and DeepSeek-R1 [8] employ explicit Chain-of-Thought (CoT) reasoning [34], achieving strong performance in complex tasks including mathematical reasoning and programming [40].

Existing studies identify three main directions for developing o1-level reasoning capabilities [6, 8, 10, 14, 32]: exposure to flawed reasoning patterns during training [37]; curriculum-based enhancement using distilled reasoning data [27]; and reinforcement learning for extended CoT capabilities [6, 8, 14]. Despite these advances, LRMs remain constrained by fixed parametric knowledge, limiting their ability to incorporate time-sensitive external information.

From a task perspective, the structured reasoning capabilities of LRMs align naturally with the requirements of TLS, which demands multi-step temporal inference, cross-document abstraction, and the ability to revise intermediate representations. Unlike prior agent-based or prompt-driven LLM approaches, which typically rely on fixed retrieval patterns or single-pass reasoning, LRMs maintain persistent reasoning states that can be incrementally updated with newly retrieved evidence. This capability is crucial for constructing timelines that are both temporally consistent and complete, as it allows the model to iteratively detect missing events, revise earlier conclusions, and integrate evidence across documents. In this way, LRMs provide a principled foundation for TimelineReasoner's reasoning-driven approach to open-domain TLS.

2.3 Deep Research

Deep Research refers to reasoning-centric frameworks that enable LLMs to autonomously conduct multi-step information acquisition and evidence-based synthesis over open and evolving knowledge sources. Unlike conventional retrieval-augmented generation, which mainly focuses on augmenting generation with external documents, Deep Research frameworks emphasize iterative reasoning loops that tightly interleave planning, targeted retrieval, verification, and memory updates [18, 22, 28].

Early examples such as WebGPT [28] decompose complex questions into search queries and synthesize answers from retrieved evidence. Recent agent-based frameworks like WebThinker [23] and

WebWeaver [24], integrate reasoning with structured retrieval, combining planning, evidence acquisition, and synthesis. Tongyi DeepResearch [19] further introduces an agentic LLM trained end-to-end with synthetic interaction data and reinforcement learning to support deep, multi-step information-seeking tasks.

Despite these advances, existing Deep Research systems focus on open-domain question-answering or report generation, and rarely target timeline-centric reasoning. Temporal dependencies are typically implicit, lacking mechanisms for detecting missing events or ensuring chronological consistency. In contrast, TimelineReasoner frames TLS as a Deep Research problem. This enables dynamic evidence acquisition, iterative refinement, and principled temporal reasoning, systematically addressing missing information and maintaining coherent event evolution in evolving news corpora.

3 Methodology

3.1 Overview

TimelineReasoner is a reasoning-driven framework for timeline summarization that treats timeline construction as an iterative reasoning and revision process rather than single-pass generation. It adopts a coarse-to-fine design that separates *global event comprehension* from *gap-driven detail refinement*, enabling the model to explicitly detect missing events and temporal inconsistencies. As illustrated in Figure 2, TimelineReasoner consists of two main stages—**Global Cognition** and **Detailed Exploration**—supported by specialized mechanisms for structured event acquisition, incremental timeline updating, and meta-level supervision, with the LRM serving as the central controller for reasoning and retrieval.

3.2 Global Cognition

When constructing a timeline, human analysts often first form a coarse-grained but comprehensive understanding of how events unfold over time before diving into the details. Inspired by this process, TimelineReasoner incorporates a **Global Cognition** stage, in which the LRM synthesizes a temporally organized summary of the evolving event landscape. This high-level overview serves as a scaffold for subsequent, more fine-grained reasoning, allowing the system to anticipate gaps, track major developments, and maintain temporal coherence from the outset.

The main output of the Global Cognition stage is the **global event memory** ξ, a structured set of temporally anchored events. Each event consists of a timestamp, a concise description, and associated entities. Importantly, ξ is not intended to be the final timeline; rather, it provides a stable, high-level representation that guides downstream reasoning in the Detailed Exploration stage. By explicitly separating global structure from local detail, this design ensures that TimelineReasoner can reason about coverage, event ordering, and missing information in a principled manner.

3.2.1 Initializing and Updating Global Memory. To construct this global event memory, TimelineReasoner combines structured evidence acquisition with reasoning in a coherent loop. Given an input query q, the **Event Scraper** retrieves relevant news documents and extracts lightweight **event metadata** m, which includes temporal markers and short descriptions. To manage large collections efficiently, articles are divided into semantically coherent chunks, each processed independently. The resulting metadata fragments are then synthesized by the LRM to produce an initial global event memory:

$$\xi^{(0)} = LRM\left(I, m\right), m = \textsc{EventScraper}(q),$$

where I is the instruction prompt guiding structured reasoning.

Crucially, the global event memory is designed to be dynamic rather than static. As new evidence is discovered during subsequent iterations, the memory is incrementally updated:

$$\xi^{(t)} = LRM\left(\xi^{(t-1)}, m^{(t)}\right),$$

where $m^{(t)}$ represents newly retrieved event metadata. This iterative update mechanism allows the global memory to remain temporally consistent, reflect newly acquired information, and provide a continuously reliable scaffold for the reasoning process. By explicitly modeling a dynamic, high-level event representation, TimelineReasoner mirrors human analysts' strategy of maintaining a constantly updated mental model of the evolving situation, ensuring that subsequent detailed exploration is guided by a coherent global perspective throughout the entire reasoning process.

Table 1: Illustrative example of event metadata and global event memory.

News Query: "Apple's pivotal product announcements"
Extracted Event Metadata (Excerpt): *January 24, 1984:* Unveiling of the Macintosh computer. *November 10, 2001:* Introduction of the iPod.
Global Event Memory ξ (Abbreviated): January 24, 1984: Unveiling of the Macintosh Computer. This event introduced the first mass-market personal computer...November 10, 2001: Introduction of the iPod. This announcement marked Apple's successful entry into the consumer electronics market....

3.3 Detailed Exploration

3.3.1 Gap-Driven Refinement. While Global Cognition provides a coarse-grained overview of the event landscape, it inevitably abstracts away fine-grained details and may miss less salient but temporally important events. To progressively refine the timeline, TimelineReasoner enters the **Detailed Exploration** stage, where the focus shifts from global structure to resolving specific informational deficiencies in a targeted way.

In this stage, TimelineReasoner main agent jointly reasons over the global event memory ξ and the current timeline memory $\mathcal{M}_{\mathcal{T}}$ to identify explicit gaps, including missing events, ambiguous or coarse timestamps, and under-specified event descriptions. Rather than exhaustively retrieving additional documents—which would introduce redundancy and noise—the framework adopts a **gap-driven strategy**, where each iteration concentrates on a small number of well-defined deficiencies. This design keeps the reasoning process targeted and allows retrieval to be directly guided by the evolving timeline state.

Based on the identified gaps, the agent formulates targeted search queries $sq^{(k)}$, which are executed by the Event Scraper to retrieve refined and temporally relevant evidence. In this way, retrieval is

tightly coupled with reasoning needs, ensuring that newly acquired information directly contributes to improving timeline completeness and temporal precision.

3.3.2 Iterative Timeline Refinement. Once sufficient evidence is obtained for the identified gaps, the agent synthesizes the retrieved information into a localized sub-timeline $\mathcal{T}^{(k)}$, capturing newly clarified or previously missing events within a limited temporal scope. Constructing sub-timelines, rather than immediately revising the entire timeline, allows TimelineReasoner to isolate local updates and reduce the risk of propagating errors across the timeline.

The generated sub-timeline is then incrementally integrated into the existing timeline memory via the **Timeline Updater**, which reconciles new evidence with previously established events and enforces consistency with the global event memory:

$$\mathcal{M}_{\mathcal{T}}^{(k)} = \textsc{TimelineUpdater}\left(\mathcal{M}_{\mathcal{T}}^{(k-1)}, \mathcal{T}^{(k)}, \xi\right).$$

This integration mechanism enables the timeline to evolve gradually through revision and refinement, rather than committing to a fixed structure in a single pass.

After each update, the **Supervisor** performs a meta-level evaluation of the current timeline memory, assessing semantic completeness, event coverage, and temporal density. This step is necessary because long reasoning chains and accumulated updates may obscure unresolved gaps or introduce subtle inconsistencies. If deficiencies remain, the Supervisor proposes a structured search plan to guide the next iteration of gap-driven reasoning and retrieval; otherwise, the process terminates. Through this iterative reasoning–retrieval–integration loop, TimelineReasoner converges to a timeline that is both comprehensive and temporally coherent.

3.4 Specialized Mechanisms

TimelineReasoner relies on several specialized mechanisms that are integral to its reasoning-driven framework. These mechanisms, including **Event Scraper**, **Timeline Updater**, and **Supervisor**, enable the LRM to iteratively acquire evidence, refine timeline memory, and enforce temporal consistency. These modules implement the core operational capabilities required for coherent and complete timeline construction, including retrieval, integration, and quality control tasks in a unified and coordinated manner.

3.4.1 Event Scraper. The Event Scraper is responsible for structured evidence acquisition from unstructured documents. News articles $A = [a_1, a_2, \ldots, a_n]$ are divided into manageable chunks c_i of size m to balance efficiency and information coverage. For each chunk, the model extracts **event metadata** including timestamps and concise descriptions:

$$event\_metadata_i = LLM\{I, c_i\},$$

where LLM denotes language model inference and I is the instruction prompt. Aggregating all chunks produces a comprehensive event metadata set that forms the basis for global event reasoning. This mechanism ensures that important events are not overlooked and provides the LRM with structured input for downstream reasoning, addressing the common TLS challenge of incomplete or fragmented event coverage.

3.4.2 Timeline Updater. The Timeline Updater implements incremental memory integration, allowing the evolving timeline to incorporate newly retrieved events while maintaining consistency with the global event memory ξ. Given the current timeline memory $\mathcal{M}_{\mathcal{T}}$ and a newly generated sub-timeline $\mathcal{T}$, the LRM produces:

$$\mathcal{M}_{\mathcal{T}} = LRM\left(I, \mathcal{M}_{\mathcal{T}}, \xi, \mathcal{T}\right).$$

This update process ensures that previously established events are preserved or refined as new evidence becomes available. By continuously revising timeline memory, the system can correct temporal inconsistencies and gradually build a coherent global timeline, addressing the TLS problem of static, single-pass generation.

3.4.3 Supervisor. The Supervisor performs meta-level reasoning over the evolving timeline memory, identifying gaps or inconsistencies that the main reasoning model may miss due to long reasoning chains or complex dependencies. It evaluates $\mathcal{M}_{\mathcal{T}}$ along three dimensions:

- **Semantic Completeness**: Ensures that events are described in sufficient detail; if not, generates a search plan to retrieve additional information.
- **Coverage Consistency**: Compares the global event memory and current timeline to detect missing events, triggering targeted retrieval when necessary.
- **Temporal Density**: Detects sparsity or event clustering in the timeline, guiding retrieval to fill temporal gaps.

Formally, the Supervisor generates a search plan as:

$$plan = LRM\left(I, \mathcal{M}_{\mathcal{T}}\right).$$

This plan guides the reasoning model in subsequent iterations, allowing TimelineReasoner to actively detect missing events, refine existing entries, and enforce chronological consistency.

4 Experiment

4.1 Implementation Details

To ensure reproducibility, all reported results are averaged over 3 independent runs with different random seeds. Experiments are designed to maintain stable outputs, and key hyperparameters are explicitly specified below.

4.1.1 Model Usage. For our method, we adopt the open-source LLM **QwQ-32B** as the main reasoning model due to its strong multi-step reasoning capabilities. The Event Scraper uses **Qwen2.5-32B** to ensure architectural consistency. Key hyperparameters for QwQ-32B are: maximum sequence length 32,768 tokens, temperature 0.7, top-p 0.9, and repetition penalty 1.05. For Qwen2.5-Instruct, the maximum sequence length is 8,192 tokens, while all other hyperparameters remain identical.

4.1.2 Search Engine. For open-domain TLS datasets, relevant documents are retrieved via Google Serper API[1]. Retrieved web pages are processed using JINA Reader[2] and converted into Markdown format for structured parsing. For closed-domain datasets such as crisis and t17, we construct an indexed news article collection using Elasticsearch [12]. Retrieval mimics realistic search behavior within

[1] https://serper.dev/
[2] https://jina.ai/reader/

Table 2: Experimental results on Open-TLS. The boldfaced score in each column denotes the best result. And the *underlined* score marks the second-highest result among the compared methods. Relative improvements (%) of TimelineReasoner over the best baseline are reported in parentheses.

Method	Model	Concat F1		Agree F1		Align F1		Date F1
		ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	
DIRECT	Qwen2.5-32B	0.2611	0.0587	0.0382	0.0131	0.0408	0.0138	0.1473
	QwQ-32B	0.2682	0.0554	0.0457	0.0142	0.0487	0.0152	0.1824
	Qwen2.5-72B	0.2908	0.0673	0.0559	0.0188	0.0613	0.0210	0.2102
REWRITE	Qwen2.5-32B	0.2779	0.0620	0.0413	0.0122	0.0447	0.0142	0.1631
	QwQ-32B	0.2862	0.0609	0.0527	0.0170	0.0580	0.0179	0.1892
	Qwen2.5-72B	0.2894	0.0740	0.0608	0.0302	0.0649	0.0308	0.1993
ITER_RAG	Qwen2.5-32B	0.3040	0.0702	0.0499	0.0180	0.0568	0.0188	0.1901
	QwQ-32B	0.2919	0.0639	0.0522	0.0183	0.0570	0.0184	0.1893
	Qwen2.5-72B	0.3292	0.0762	0.0579	0.0201	0.0633	0.0211	0.1902
CHRONOS	Qwen2.5-32B	0.3121	0.0760	0.0893	0.0311	0.0909	0.0301	0.3119
	QwQ-32B	0.3219	0.0752	0.0901	0.0292	0.0931	0.0292	0.3030
	Qwen2.5-72B	0.3380	0.0801	0.0962	0.0302	0.0991	0.0332	0.3320
TimelineReasoner	QwQ-32B	**0.3631** (+7.4%)	**0.0894** (+11.6%)	**0.1130** (+17.5%)	**0.0400** (+28.6%)	**0.1233** (+24.4%)	**0.0411** (+23.8%)	**0.3765** (+13.4%)

Table 3: Closed-domain TLS performance comparison. The boldfaced score indicates the highest result, while the *underlined* score marks the second-highest result among the compared methods. Relative improvements (%) of TimelineReasoner over the best baseline are reported in parentheses.

Method	Model	Crisis			T17		
		Align R-1	Align R-2	Date F1	Align R-1	Align R-2	Date F1
CLUST		0.0610	0.0131	0.2260	0.0822	0.0201	0.4070
EGC		0.0791	0.0149	0.2911	**0.1031**	0.0239	**0.5501**
LLM-TLS	Qwen2.5-32B	0.0701	0.0199	0.2732	0.0878	0.0204	0.5157
	QwQ-32B	0.0678	0.0189	0.2700	0.0852	0.0178	0.5133
DIRECT	Qwen2.5-32B	0.0162	0.0040	0.0639	0.0537	0.0132	0.3408
	QwQ-32B	0.0184	0.0020	0.0678	0.0510	0.0140	0.3064
REWRITE	Qwen2.5-32B	0.0342	0.0190	0.0927	0.0568	0.0160	0.3619
	QwQ-32B	0.0441	0.0243	0.1030	0.0552	0.0139	0.3492
ITER_RAG	Qwen2.5-32B	0.0552	0.0130	0.1279	0.0682	0.0201	0.3810
	QwQ-32B	0.0579	0.0173	0.2201	0.0848	0.0282	0.4008
CHRONOS	Qwen2.5-32B	0.0861	0.0295	0.2583	0.0811	0.0220	0.3822
	QwQ-32B	0.0828	0.0256	0.2907	0.0901	0.0234	0.4730
TimelineReasoner	QwQ-32B	**0.1080** (+25.4%)	**0.0331** (+12.2%)	**0.3509** (+20.5%)	0.1012 (-1.8%)	**0.0322** (+14.2%)	0.5453 (-0.9%)

the closed corpus. In both types of datasets, we retrieve the top_k (k=20) documents for each query.

4.2 Evaluation Metrics

We evaluate generated timelines against reference timelines using the Tilse framework [25, 26], which is specifically designed for TLS.

4.2.1 ROUGE-N. We report ROUGE-1 and ROUGE-2 F1 scores to measure lexical overlap between generated and reference timelines. Following Tilse, we consider three variants: (1) *Concat F1* computes ROUGE by concatenating all date summaries to assess overall content overlap; (2) *Agree F1* calculates ROUGE only for summaries of matching dates, focusing on precision; and (3) *Align F1* aligns predicted and reference summaries based on temporal and semantic distance, then computes ROUGE with penalties for distant alignments.

4.2.2 Date F1-score (Date-F1). Date-F1 measures the accuracy of date prediction by computing the F1 score between generated and reference event dates, evaluating temporal coverage and alignment.

4.3 Open-domain TLS

4.3.1 Baseline. We compare TimelineReasoner with four representative baselines for open-domain TLS. All methods retrieve the

Table 4: Impact of retrieval scale (N) on Open-TLS performance. We evaluate the impact of the number of documents in the Initialization of Global Event Memory (N_{init}) and the Detailed Exploration (N_{exp}). Bold indicates the best, *underlined* indicates the second best.

Parameter	N	Concat F1		Agree F1		Align F1		Date F1
		ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	
N_{init}	10	0.3333	0.0739	0.0811	0.0260	0.0926	0.0331	0.3336
	20	**0.3631**	**0.0884**	**0.1130**	**0.0400**	**0.1233**	**0.0411**	0.3764
	30	0.3357	0.0768	0.0966	0.0322	0.1035	0.0358	**0.3777**
	40	0.3434	0.0821	0.0992	0.0346	0.1068	0.0362	0.3535
N_{exp}	10	0.3361	0.0759	0.0933	0.0277	0.1050	0.0298	0.3352
	20	**0.3631**	**0.0884**	**0.1130**	**0.0400**	**0.1233**	**0.0411**	**0.3764**
	30	0.3357	0.0792	0.0949	0.0309	0.1058	0.0328	0.3534
	40	0.3134	0.0699	0.0859	0.0264	0.0969	0.0285	0.3513

same number of documents to ensure fair comparison. (1) DIRECT: retrieves documents using the original query and generates the timeline in a single pass. (2) REWRITE: expands the original query into 2-3 rewritten variants, aggregates retrieved documents, and generates the timeline. (3) ITER_RAG: performs iterative retrieval over five rounds, where each round refines the query based on the current timeline and incrementally updates the output. (4) CHRONOS [35]: an open-domain TLS framework based on iterative self-questioning to collect multi-perspective event information. For CHRONOS, we replace the original search module with Google Serper to unify the retrieval interface across methods, while keeping other settings unchanged. All baselines are evaluated using Qwen2.5-72B, Qwen2.5-32B, and QwQ-32B to control for backbone effects.

4.3.2 Result Analysis. As shown in Table 2, TimelineReasoner consistently outperforms all baselines across all evaluation metrics. Methods incorporating query rewriting or iterative retrieval substantially outperform direct generation, confirming the importance of multi-step retrieval in open-domain TLS. Among non-proposed baselines, Qwen2.5-72B generally achieves the strongest performance, reflecting the benefits of larger model capacity. Notably, QwQ-32B performs on par with or better than Qwen2.5-32B despite similar parameter scales, indicating the advantage of enhanced reasoning capability for timeline construction. TimelineReasoner achieves statistically significant gains in both ROUGE-N and Date-F1, demonstrating improved content coverage and more accurate temporal alignment. These results highlight the effectiveness of reasoning-driven, gap-aware refinement for open-domain timeline summarization. Overall, TimelineReasoner demonstrates strong robustness and effectiveness in open-domain TLS by producing timelines that are both information-rich and temporally accurate.

4.4 Closed-domain TLS

4.4.1 Baseline. We compare our method with representative closed-domain TLS approaches, including traditional and LLM-based methods. Specifically, we include: (1) CLUST [11], a clustering-based approach using temporal frequency-weighted event aggregation; (2) EGC [21], a graph-based method that selects salient events via time-aware optimal transport; (3) LLM-TLS [15], a LLM-driven framework that employs LLMs as pseudo-oracles for incremental event clustering in streaming data. In addition, we evaluate several strong LLM-based baselines originally designed for open-domain TLS, including DIRECT, REWRITE, ITER_RAG, and CHRONOS, to assess their generalization to closed-domain settings. For fair comparison, all model-based baselines are implemented using Qwen2.5-32B and QwQ-32B, consistent with our framework.

Table 5: Comparison of token consumption and performance (Align ROUGE-2) across TLS methods

Method	CHRONOS	TimelineReasoner	LLM-TLS
Token	~0.7M	~0.9M	~47M
Performance	0.0295	**0.0331**	0.0199

4.4.2 Result Analysis. We evaluate our method on two widely used benchmarks, *Crisis* and *T17*, using Aligned F1 (short for AR-1/AR-2) and Date F1 as evaluation metrics. Baseline configurations are aligned with prior work, and minor variations may arise from differences in document segmentation. The experimental results are reported in Table 3. Overall, the results demonstrate strong cross-domain generalization from open-domain to closed-domain TLS. On the *Crisis* dataset, our method outperforms all competing approaches, including CHRONOS, traditional methods, as well as the LLM-based LLM-TLS. On *T17*, our approach achieves the best performance on AR-2, while ranking second on other metrics. These results indicate that our framework maintains robust performance across datasets with varying topical scopes and temporal distributions. Furthermore, the consistent improvements observed in both content-oriented metrics and temporal accuracy suggest that our method can effectively capture salient event information while preserving chronological structure. Overall, these findings confirm that our reasoning-driven framework remains effective when applied to closed-domain settings with fixed document collections.

4.4.3 Limitations. While TimelineReasoner achieves strong performance in closed-domain TLS, its advantages are most pronounced in settings that require iterative evidence acquisition and timeline revision. In closed-domain benchmarks with high document redundancy and clearly defined event structures, the benefits of

Table 6: Ablation study conducted on Open-TLS. The boldfaced score indicates the highest result.

Method	Concat F1		Agree F1		Align F1		Date F1
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	
Ours	**0.363**	**0.089**	**0.113**	**0.040**	**0.123**	**0.041**	**0.376**
w/o Update Global Event Memory	0.330	0.073	0.088	0.027	0.095	0.032	0.338
w/o Global Event Memory	0.298	0.055	0.038	0.011	0.045	0.012	0.154
w/o Timeline Updater	0.318	0.075	0.088	0.028	0.094	0.030	0.339
w/o Supervisor	0.338	0.082	0.102	0.036	0.111	0.037	0.360

reasoning-driven refinement may be less prominent compared to open-domain scenarios. Addressing retrieval and reasoning strategies that are better tailored to highly redundant corpora remains an interesting direction for future work.

4.5 Cost Analysis

Cost and Token Efficiency. To assess the cost-efficiency of TimelineReasoner, we compare the total token consumption of different TLS approaches under a closed-domain setting, where the document collection is fixed and shared across methods. This controlled setup enables a fair and reproducible comparison by eliminating variability introduced by open-domain retrieval, such as fluctuating document availability and query-dependent retrieval depth.

As shown in Table 5, TimelineReasoner incurs a modest increase in token usage compared to CHRONOS, reflecting the additional reasoning steps involved in global event abstraction and coherence planning. However, it is substantially more token-efficient than LLM-TLS, which relies on long-context processing over large document collections and consequently exhibits significantly higher token consumption. These results suggest that, despite multiple LRM invocations, TimelineReasoner achieves a favorable balance between reasoning capability and computational cost by leveraging structured reasoning and selective memory construction rather than brute-force long-context generation.

4.6 Ablation Study

4.6.1 Impact of Retrieval Scale. We evaluate the sensitivity of the retrieval scale by varying the documents retrieved for Global Cognition (N_{init}) and Detail Exploration (N_{exp}) within $\{10, 20, 30, 40\}$. As shown in Table 4, $N = 20$ for both stages provides the optimal balance between performance and efficiency. While $N_{init} = 10$ establishes a basic event skeleton, increasing it to 20 enhances evidence diversity and ROUGE scores. However, $N_{init} = 40$ leads to a performance plateau, suggesting that excessive initial data introduces redundancy that complicates the LRM's construction of global event memory. Notably, performance is more sensitive to N_{exp}. Insufficient retrieval ($N_{exp} = 10$) hinders the Supervisor from identifying information gaps, while excessive documents ($N_{exp} > 20$) introduce noise and conflicting reports, distracting the LRM during iterative reasoning. Consequently, $N = 20$ serves as a robust threshold, ensuring both a coherent global perspective and high-fidelity detail refinement.

4.6.2 Role of Global Event Memory. Global event memory maintains a unified representation of event information and provides persistent guidance for both the main reasoning process and the supervisor. As shown in Table 6, removing global event memory leads to substantial performance degradation across all metrics, with notable drops in both Date F1 and ROUGE scores. These results indicate that global event memory is indispensable for preserving temporal accuracy and modeling inter-event relationships, serving simultaneously as a knowledge repository and a coordination mechanism for reasoning.

Table 7: Performance comparison under different backbone models on Open-TLS. The best and second-best results are in bold and *underlined*. DS-V3.2 is short for DeepSeek-V3.2.

LLM	Concat F1		Agree F1		Align F1		Date F1
	R-1	R-2	R-1	R-2	R-1	R-2	
CHRONOS							
QwQ-32B	0.322	0.075	0.090	0.029	0.093	0.029	0.303
Qwen2.5-72B	0.338	0.080	0.096	0.030	0.099	0.033	0.332
DS-V3.2	0.359	0.089	0.097	0.036	0.104	0.038	0.351
Ours							
QwQ-32B	0.363	0.089	0.113	0.040	0.123	0.041	0.376
Qwen2.5-72B	0.326	0.031	0.077	0.030	0.083	0.032	0.240
DS-V3.2	**0.378**	**0.106**	**0.141**	**0.054**	**0.149**	**0.055**	**0.410**

We further examine the impact of memory maintenance by disabling updates with event metadata after each Event Scraper call. This ablation leads to consistent but smaller performance decreases, suggesting that continuous memory refinement helps maintain an up-to-date and comprehensive event representation, which in turn supports more stable reasoning and timeline generation.

4.6.3 Impact of Timeline Memory. Timeline Updater serves a critical function in refining the timeline memory within our framework. When this component is removed, the system falls back to naive concatenation of subtimelines. As reported in Table 6, this modification leads to consistent performance degradation across all evaluation metrics, indicating reduced quality in both event description granularity and temporal coverage precision. These results confirm the updater's essential role in memory optimization, where it actively synthesizes and reorganizes temporal information rather than merely aggregating raw outputs. The observed performance drop, while moderate in scale, systematically demonstrates how Timeline Updater's hierarchical processing of temporal relationships contributes to final timeline coherence.

4.6.4 Importance of Supervisor. The Supervisor monitors the evolving timeline memory and identifies potential deficiencies after the

Table 8: Case study of the process of TimelineReasoner generating a timeline.

Reasoning chain of TimelineReasoner
You are a reasoning assistant with the ability to perform web searches to help create an accurate chronological timeline. ... The event development process: ... Now, proceed to make the necessary search queries to confirm these points and ensure accuracy. First, verifying the Mac Studio M2 Ultra's announcement date: <|begin_search_query|>Apple Mac Studio M2 Ultra release date<|end_search_query|> (Here call Event Scraper) ... Let me structure each part clearly and ensure the descriptions are detailed as per the search results. <|begin_update_timeline|> ... <|end_update_timeline|> (Here call Timeline Updater) ... Wait, I think the current timeline is not comprehensive enough. I need to search for more information. Here is my future search plan: ... (Here Supervisor returns a search plan) ...

main reasoning model updates the timeline memory. It then generates targeted search plans to guide subsequent retrieval, helping focus the reasoning process and avoid unnecessary computation. As shown in Table 6, removing the Supervisor leads to consistent performance drops across all metrics, with particularly noticeable decreases in temporal coherence and event coverage. These results suggest that the Supervisor contributes to more accurate and complete timeline construction by coordinating iterative reasoning and targeted retrieval, without which the system's ability to detect and fill gaps is diminished.

4.7 Backbone Robustness Analysis

To verify the capability of our framework to effectively exploit the reasoning capacity of large reasoning models, we evaluate multiple backbone models beyond the default QwQ-32B used in our method. Specifically, we replace the backbone with an instruction model, Qwen2.5-72B, as well as a recently released model with strong reasoning and agent capabilities, DeepSeek-V3.2. All experiments are conducted on the Open-TLS benchmark, and the results are reported in Table 7. The results show that, within our framework, the two reasoning models consistently outperform Qwen2.5-72B across multiple evaluation metrics, indicating that our method can effectively leverage enhanced reasoning abilities. Moreover, when comparing reasoning backbones, DeepSeek-V3.2 achieves clear performance gains over QwQ-32B, suggesting that stronger reasoning backbones further amplify the effectiveness of our approach. In contrast, when applied to the baseline, DeepSeek-V3.2 does not demonstrate a comparable advantage: its performance is substantially lower than that achieved by DeepSeek-V3.2 within our framework, and is roughly on par with QwQ-32B when used as the backbone of our method. These findings demonstrate that our method not only benefits from stronger reasoning backbones, but also more effectively translates the reasoning capacity of large models into improved performance for complex timeline summarization tasks.

4.8 Case Study

To illustrate how TimelineReasoner performs reasoning-driven timeline construction, we present a case study on the news query *Apple's pivotal product announcements*, as shown in Table 8. Starting from the global event memory, the reasoning model identifies a missing temporal detail for a salient event (the announcement date of the Mac Studio M2 Ultra). To resolve this gap, it formulates a targeted query, which is executed by the Event Scraper to retrieve temporally relevant evidence. The retrieved information is synthesized into a localized sub-timeline and incrementally integrated into the existing timeline via the Timeline Updater, refining the timeline without altering previously confirmed events. After the update, the Supervisor evaluates the timeline and detects remaining coverage deficiencies, generating a structured search plan to guide further reasoning and retrieval. This iterative process continues until the timeline satisfies completeness and temporal consistency criteria. This case study highlights how TimelineReasoner explicitly detects missing information, performs gap-driven retrieval, and revises the timeline through iterative reasoning.

5 Conclusion and Future Work

In this work, we propose TimelineReasoner, an active, reasoning-driven framework for TLS. By leveraging the structured reasoning capabilities of LRMs, our framework decouples macroscopic event tracking from fine-grained detail refinement through a two-stage paradigm: Global Cognition and Detail Exploration. Supported by specialized mechanisms, TimelineReasoner maintains a dynamic timeline memory that is iteratively refined with targeted evidence. Extensive evaluations demonstrate that TimelineReasoner significantly outperforms LLM-based baselines in open-domain TLS tasks and remains highly competitive in closed-domain settings. Our analysis further reveals that the framework is uniquely capable of translating the latent reasoning power of LRMs into tangible performance gains while maintaining a favorable balance between output fidelity and token efficiency.

In the future, we plan to explore other ways to enhance the performance of our method. First, we plan to apply reinforcement learning to develop specialized reward functions, optimizing the specificity of generated queries and the strategic quality of search plans. Second, we will investigate adaptive reasoning strategies tailored for redundant information environments. Finally, we aim to extend the framework's memory scaling capabilities to support the construction of comprehensive timelines for event sequences spanning multiple years.

Acknowledgments

This work was supported by National Natural Science Foundation of China No. 62272467. The work was partially done at the Beijing Key Laboratory of Research on Large Models and Intelligent Governance and Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE.

References

[1] James Allan, Rahul Gupta, and Vikas Khandelwal. 2001. Temporal Summaries of News Topics. In *SIGIR 2001: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA*, W. Bruce Croft, David J. Harper, Donald H. Kraft, and Justin Zobel (Eds.). ACM, 10–18. doi:10.1145/383952.383954

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen Technical Report. *CoRR* abs/2309.16609 (2023). arXiv:2309.16609 doi:10.48550/ARXIV.2309.16609

[3] Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2025. Towards Reasoning Era: A Survey of Long Chain-of-Thought for Reasoning Large Language Models. *CoRR* abs/2503.09567 (2025). arXiv:2503.09567 doi:10.48550/ARXIV.2503.09567

[4] Xiuying Chen, Zhangming Chan, Shen Gao, Meng-Hsuan Yu, Dongyan Zhao, and Rui Yan. 2019. Learning towards Abstractive Timeline Summarization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, Sarit Kraus (Ed.). ijcai.org, 4939–4945. doi:10.24963/IJCAI.2019/686

[5] Xiuying Chen, Mingzhe Li, Shen Gao, Zhangming Chan, Dongyan Zhao, Xin Gao, Xiangliang Zhang, and Rui Yan. 2023. Follow the Timeline! Generating an Abstractive and Extractive Timeline Summary in Chronological Order. *ACM Trans. Inf. Syst.* 41, 1 (2023), 9:1–9:30. doi:10.1145/3517221

[6] Zhipeng Chen, Yingqian Min, Beichen Zhang, Jie Chen, Jinhao Jiang, Daixuan Cheng, Wayne Xin Zhao, Zheng Liu, Xu Miao, Yang Lu, Lei Fang, Zhongyuan Wang, and Ji-Rong Wen. 2025. An Empirical Study on Eliciting and Improving R1-like Reasoning Models. *CoRR* abs/2503.04548 (2025). arXiv:2503.04548 doi:10.48550/ARXIV.2503.04548

[7] Hai Leong Chieu and Yoong Keok Lee. 2004. Query based event extraction along a timeline. In *SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, July 25-29, 2004*, Mark Sanderson, Kalervo Järvelin, James Allan, and Peter Bruza (Eds.). ACM, 425–432. doi:10.1145/1008992.1009065

[8] DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *CoRR* abs/2501.12948 (2025). arXiv:2501.12948 doi:10.48550/ARXIV.2501.12948

[9] Yijun Duan, Adam Jatowt, and Masatoshi Yoshikawa. 2020. Comparative Timeline Summarization via Dynamic Affinity-Preserving Random Walk. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020) (Frontiers in Artificial Intelligence and Applications, Vol. 325)*, Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang (Eds.). IOS Press, 1778–1785. doi:10.3233/FAIA200292

[10] Mohamed Amine Ferrag, Norbert Tihanyi, and Mérouane Debbah. 2025. Reasoning beyond limits: Advances and open problems for LLMs. *ICT Express* 11, 6 (2025), 1054–1096. doi:10.1016/J.ICTE.2025.09.003

[11] Demian Gholipour Ghalandari and Georgiana Ifrim. 2020. Examining the State-of-the-Art in News Timeline Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (Eds.). Association for Computational Linguistics, 1322–1334. doi:10.18653/V1/2020.ACL-MAIN.122

[12] Clinton Gormley and Zachary J. Tong. 2015. Elasticsearch: The Definitive Guide. https://api.semanticscholar.org/CorpusID:62964734

[13] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B. Brown, Prafulla Dhariwal, Scott Gray, Chris Hallacy, Benjamin Mann, Alec Radford, Aditya Ramesh, Nick Ryder, Daniel M. Ziegler, John Schulman, Dario Amodei, and Sam McCandlish. 2020. Scaling Laws for Autoregressive Generative Modeling. *CoRR* abs/2010.14701 (2020). arXiv:2010.14701 https://arxiv.org/abs/2010.14701

[14] Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. 2025. Open-Reasoner-Zero: An Open Source Approach to Scaling Up Reinforcement Learning on the Base Model. *CoRR* abs/2503.24290 (2025). arXiv:2503.24290 doi:10.48550/ARXIV.2503.24290

[15] Qisheng Hu, Geonsik Moon, and Hwee Tou Ng. 2024. From Moments to Milestones: Incremental Timeline Summarization Leveraging Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, 7232–7246. doi:10.18653/V1/2024.ACL-LONG.390

[16] 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, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, and Ilge Akkaya. 2024. OpenAI o1 System Card. *CoRR* abs/2412.16720 (2024). arXiv:2412.16720 doi:10.48550/ARXIV.2412.16720

[17] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large Language Models are Zero-Shot Reasoners. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/8bb0d291acd4acf06ef112099c16f326-Abstract-Conference.html

[18] Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-Augmented Dialogue Generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, 8460–8478. doi:10.18653/V1/2022.ACL-LONG.579

[19] Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, Xixi Wu, Xuanzhong Chen, Yida Zhao, Zhen Zhang, Zhengwei Tao, Zhongwang Zhang, Zile Qiao, Chenxi Wang, Donglei Yu, Gang Fu, Haiyang Shen, Jiayin Yang, Jun Lin, Junkai Zhang, Kui Zeng, Li Yang, Hailong Yin, Maojia Song, Ming Yan, Peng Xia, Qian Xiao, Rui Min, Ruixue Ding, Runnan Fang, Shaowei Chen, Shen Huang, Shihang Wang, Shihao Cai, Weizhou Shen, Xiaobin Wang, Xin Guan, Xinyu Geng, Yingcheng Shi, Yuning Wu, Zhuo Chen, Zijian Li, and Yong Jiang. 2025. Tongyi DeepResearch Technical Report. *CoRR* abs/2510.24701 (2025). arXiv:2510.24701 doi:10.48550/ARXIV.2510.24701

[20] Jiwei Li and Sujian Li. 2013. Evolutionary Hierarchical Dirichlet Process for Timeline Summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*. The Association for Computer Linguistics, 556–560. https://aclanthology.org/P13-2099/

[21] Manling Li, Tengfei Ma, Mo Yu, Lingfei Wu, Tian Gao, Heng Ji, and Kathleen R. McKeown. 2021. Timeline Summarization based on Event Graph Compression via Time-Aware Optimal Transport. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, 6443–6456. doi:10.18653/V1/2021.EMNLP-MAIN.519

[22] Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic Search-Enhanced Large Reasoning Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, 5420–5438. doi:10.18653/V1/2025.EMNLP-MAIN.276

[23] Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yutao Zhu, Yongkang Wu, Ji-Rong Wen, and Zhicheng Dou. 2025. WebThinker: Empowering Large Reasoning Models with Deep Research Capability. *CoRR* abs/2504.21776 (2025). arXiv:2504.21776 doi:10.48550/ARXIV.2504.21776

[24] Zijian Li, Xin Guan, Bo Zhang, Shen Huang, Houquan Zhou, Shaopeng Lai, Ming Yan, Yong Jiang, Pengjun Xie, Fei Huang, Jun Zhang, and Jingren Zhou. 2025. WebWeaver: Structuring Web-Scale Evidence with Dynamic Outlines for Open-Ended Deep Research. *CoRR* abs/2509.13312 (2025). arXiv:2509.13312 doi:10.48550/ARXIV.2509.13312

[25] Sebastian Martschat and Katja Markert. 2017. Improving ROUGE for Timeline Summarization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, Mirella Lapata, Phil Blunsom, and Alexander Koller (Eds.). Association for Computational Linguistics, 285–290. doi:10.18653/V1/E17-2046

[26] Sebastian Martschat and Katja Markert. 2018. A Temporally Sensitive Submodularity Framework for Timeline Summarization. In *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, Anna Korhonen and Ivan Titov (Eds.). Association for Computational Linguistics, 230–240. doi:10.18653/V1/K18-1023

[27] Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, Wayne Xin Zhao, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. 2024. Imitate, Explore, and Self-Improve: A Reproduction Report on Slow-thinking Reasoning Systems. *CoRR* abs/2412.09413 (2024). arXiv:2412.09413 doi:10.48550/ARXIV.2412.09413

[28] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. WebGPT: Browser-assisted question-answering with human feedback. *CoRR* abs/2112.09332 (2021). arXiv:2112.09332 https://arxiv.org/abs/2112.09332

[29] Kiem-Hieu Nguyen, Xavier Tannier, and Véronique Moriceau. 2014. Ranking Multidocument Event Descriptions for Building Thematic Timelines. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, Jan Hajic and Junichi Tsujii (Eds.). ACL, 1208–1217. https://aclanthology.org/C14-1114/

[30] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023). arXiv:2303.08774 doi:10.48550/ARXIV.2303.08774

[31] Julius Steen and Katja Markert. 2019. Abstractive Timeline Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, Lu Wang, Jackie Chi Kit Cheung, Giuseppe Carenini, and Fei Liu (Eds.). Association for Computational Linguistics, Hong Kong, China, 21–31. doi:10.18653/v1/D19-5403

[32] Xinyu Tang, Xiaolei Wang, Zhihao Lv, Yingqian Min, Xin Zhao, Binbin Hu, Ziqi Liu, and Zhiqiang Zhang. 2025. Unlocking General Long Chain-of-Thought Reasoning Capabilities of Large Language Models via Representation Engineering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (Eds.). Association for Computational Linguistics, 6832–6849. https://aclanthology.org/2025.acl-long.339/

[33] Qwen Team. 2025. QwQ-32B: Embracing the Power of Reinforcement Learning. https://qwenlm.github.io/blog/qwq-32b/

[34] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

[35] Weiqi Wu, Shen Huang, Yong Jiang, Pengjun Xie, Fei Huang, and Hai Zhao. 2025. Unfolding the Headline: Iterative Self-Questioning for News Retrieval and Timeline Summarization. In *Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, Luis Chiruzzo, Alan Ritter, and Lu Wang (Eds.). Association for Computational Linguistics, 4385–4398. doi:10.18653/V1/2025.FINDINGS-NAACL.248

[36] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 Technical Report. *CoRR* abs/2412.15115 (2024). arXiv:2412.15115 doi:10.48550/ARXIV.2412.15115

[37] Tian Ye, Zicheng Xu, Yuanzhi Li, and Zeyuan Allen-Zhu. 2025. Physics of Language Models: Part 2.2, How to Learn From Mistakes on Grade-School Math Problems. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=zpDGwcmMV4

[38] Jingyi You, Dongyuan Li, Hidetaka Kamigaito, Kotaro Funakoshi, and Manabu Okumura. 2022. Joint Learning-based Heterogeneous Graph Attention Network for Timeline Summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz (Eds.). Association for Computational Linguistics, 4091–4104. doi:10.18653/V1/2022.NAACL-MAIN.301

[39] Yi Yu, Adam Jatowt, Antoine Doucet, Kazunari Sugiyama, and Masatoshi Yoshikawa. 2021. Multi-TimeLine Summarization (MTLS): Improving Timeline Summarization by Generating Multiple Summaries. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, 377–387. doi:10.18653/V1/2021.ACL-LONG.32

[40] Duzhen Zhang, Zhong-Zhi Li, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Xiuyi Chen, Yingying Zhang, Fei Yin, Jiahua Dong, Zhijiang Guo, Le Song, and Cheng-Lin Liu. 2026. From System 1 to System 2: A Survey of Reasoning Large Language Models. *IEEE Trans. Pattern Anal. Mach. Intell.* 48, 3 (2026), 3335–3354. doi:10.1109/TPAMI.2025.3637037

[41] Wayne Xin Zhao, Yanwei Guo, Rui Yan, Yulan He, and Xiaoming Li. 2013. Timeline generation with social attention. In *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013*, Gareth J. F. Jones, Paraic Sheridan, Diane Kelly, Maarten de Rijke, and Tetsuya Sakai (Eds.). ACM, 1061–1064. doi:10.1145/2484028.2484103