DynaQuest: A Dynamic Question Answering Dataset Reflecting Real-World Knowledge Updates

Qian Lin **Junyi Li** **Hwee Tou Ng**
Department of Computer Science, National University of Singapore
qlin@u.nus.edu junyi_cs@nus.edu.sg nght@comp.nus.edu.sg

Abstract

The rapidly changing nature of real-world information presents challenges for large language models (LLMs), which are typically trained on static datasets. This limitation makes it difficult for LLMs to accurately perform tasks that require up-to-date knowledge, such as time-sensitive question answering (QA). In this paper, we introduce **DynaQuest**, a **Dyna**mic **Quest**ion answering dataset reflecting knowledge updates in the real world. DynaQuest is based on Wikipedia Infoboxes, which are frequently updated to reflect real-world changes. Our dataset is created by automatically identifying and comparing changes between different versions of Wikipedia pages and generating question-answer pairs based on these updates. To address the challenges posed by our dynamic dataset, we propose **CARL**, a **C**ontext-**A**ware **R**einforcement **L**earning framework to improve the performance of LLMs on time-sensitive question answering. We conduct experiments on our collected dataset across recent time periods and demonstrate the effectiveness of our approach. Furthermore, we maintain a dynamic knowledge updating process, providing a periodically evolving benchmark to continually evaluate LLMs' ability to answer time-sensitive questions.[1]

1 Introduction

Large language models (LLMs) have demonstrated strong capabilities in encoding extensive amounts of knowledge from massive training datasets (Radford et al., 2019; Brown et al., 2020; Touvron et al., 2023a). These models are widely applied in knowledge-intensive tasks such as question answering (QA) and reasoning, where they utilize stored world knowledge to generate appropriate answers to given queries. However, despite their strengths, LLMs face limitations when addressing time-sensitive questions, where knowledge evolves over time. The static nature of LLMs' knowledge, derived from training on fixed datasets, prevents them from integrating real-time updates, leading to inaccurate or outdated responses as real-world information changes.

Several datasets have been proposed to study time-related questions (Chen et al., 2021; Dhingra et al., 2022; Tan et al., 2023, 2024). TimeQA (Chen et al., 2021) includes temporal questions derived from Wikidata and formulates the QA task in an open-book setting. TempReason (Tan et al., 2023) and Complex-TR (Tan et al., 2024) also primarily consist of temporal questions derived from a static Wikidata corpus. These datasets lack questions to reflect real-time updates, as they rely on a fixed snapshot of knowledge. Recent works have also utilized Wikipedia articles to create time-sensitive QA datasets. GrowOver (Ko et al., 2024) and EvolvingQA (Kim et al., 2024) generate QA pairs by detecting changes in text between different versions of Wikipedia pages. While these approaches are valuable for studying time-sensitive QA, their reliance on full-text comparisons often struggles to identify precise and relevant updates, as article edits can be unstructured and noisy.

Recent dynamic QA datasets such as RealtimeQA (Kasai et al., 2023), FreshQA (Vu et al., 2023) and CDQA (Xu et al., 2025) have been introduced to address the challenges of dynamically changing knowledge. RealtimeQA (Kasai et al., 2023) collects questions and answers from news quizzes regularly published by a small number of news websites. The dataset captures evolving knowledge from current events but is limited by its reliance on the publication schedules and content preferences of selected websites, leading to restricted domain coverage. FreshQA (Vu et al., 2023) consists of natural question-answer pairs written by human annotators. Similarly, CDQA (Xu et al., 2025) focuses on dynamic QA but is

[1]Our dataset and source code are available at https://github.com/nusnlp/DynaQuest.

Findings of the Association for Computational Linguistics: ACL 2025, pages 26918–26936

specifically designed for the Chinese language. Both datasets rely on human annotators for regular answer updates, ensuring high-quality annotations. However, this labor-intensive approach limits scalability in both question volume and scope diversity.

To address the challenge of evolving world knowledge and aforementioned drawbacks, we aim to construct a dynamic time-sensitive QA dataset to reflect real-world change in time and serve as a testbed to evaluate the model's knowledge and its ability to integrate with recent knowledge updates. We propose **DynaQuest**, a novel dynamic QA dataset consisting of questions reflecting knowledge updates over time, for evaluating LLMs over rapidly changing real-world knowledge. We introduce an automated pipeline for dataset construction, enabling regular, accurate and efficient updating of questions and answers. We leverage Wikipedia Infoboxes as an efficient knowledge source for building time-sensitive QA datasets. Infoboxes offer structured and concise summaries of key information, allowing precise identification of changes while reducing computational overhead. Focusing on Infoboxes instead of passages addresses scalability challenges in existing datasets, enables regular updates, and provides a reliable testbed for evaluating LLMs on real-time knowledge integration. Our dataset provides a periodically evolving testbed to continually evaluate LLMs' ability to answer time-sensitive questions.

Furthermore, we conduct experiments on our constructed datasets using recent state-of-the-art LLMs. The results show that without access to retrieved up-to-date knowledge, even advanced LLMs struggle to perform well, reflecting the challenges posed by the dynamic and time-sensitive nature of our datasets. To address this challenge, we propose **C**ontext-**A**ware **R**einforcement **L**earning (**CARL**), a learning approach that refines how the model utilizes retrieved knowledge to solve dynamic knowledge updating. CARL optimizes the balance between retrieved and parametric knowledge, guiding the model to selectively incorporate time-sensitive information while mitigating model bias towards either retrieved context or parametric knowledge. By incorporating a reward mechanism that adjusts knowledge integration, CARL enhances the model's ability to align with evolving real-world knowledge in retrieval-augmented generation settings. Experimental results across multiple dataset periods indicate that CARL achieves performance improvements over strong baselines, demonstrating its capability to address evolving knowledge and improve LLM outputs for real-world, time-sensitive QA tasks.

In summary, our contributions are as follows:

- We propose a dynamic question answering dataset, **DynaQuest**, based on knowledge changes in Wikipedia Infoboxes to evaluate LLMs' knowledge and their ability to integrate knowledge updates over time. Our data collection pipeline enables regular, accurate, and efficient updates to both questions and answers.

- We propose a novel context-aware reinforcement learning method to enhance the model's ability to integrate retrieved knowledge while optimizing its reliance on both retrieved and parametric knowledge. Experimental results demonstrate that our method improves alignment with evolving real-world knowledge and achieves notable performance gains over strong baselines.

2 Dataset

In this section, we introduce DynaQuest, a novel dynamic dataset for evaluating LLMs on rapidly changing real-world information by constructing questions that reflect knowledge updates over time. We present the construction process of DynaQuest in Figure 1.

2.1 Source Selection

Wikipedia has become a reliable and frequently updated knowledge source for most prior work (Kim et al., 2024; Ko et al., 2024). Different from previous work utilizing unstructured passages, we consider the Infobox in Wikipedia, which is one of the most informative parts for a Wikipedia page and offers structured summaries of key facts about the subjects in this page. The infobox consists of attributes and values that highlight important facts, making them a valuable resource for identifying and tracking knowledge updates. Infobox updates reflect the latest real-world knowledge developments, further enhancing their worth for building dynamically updating datasets.

By leveraging the structured nature of Infobox, we aim to create an efficient and scalable approach for generating question-answer pairs that align with real-time knowledge updates. This focus allows us

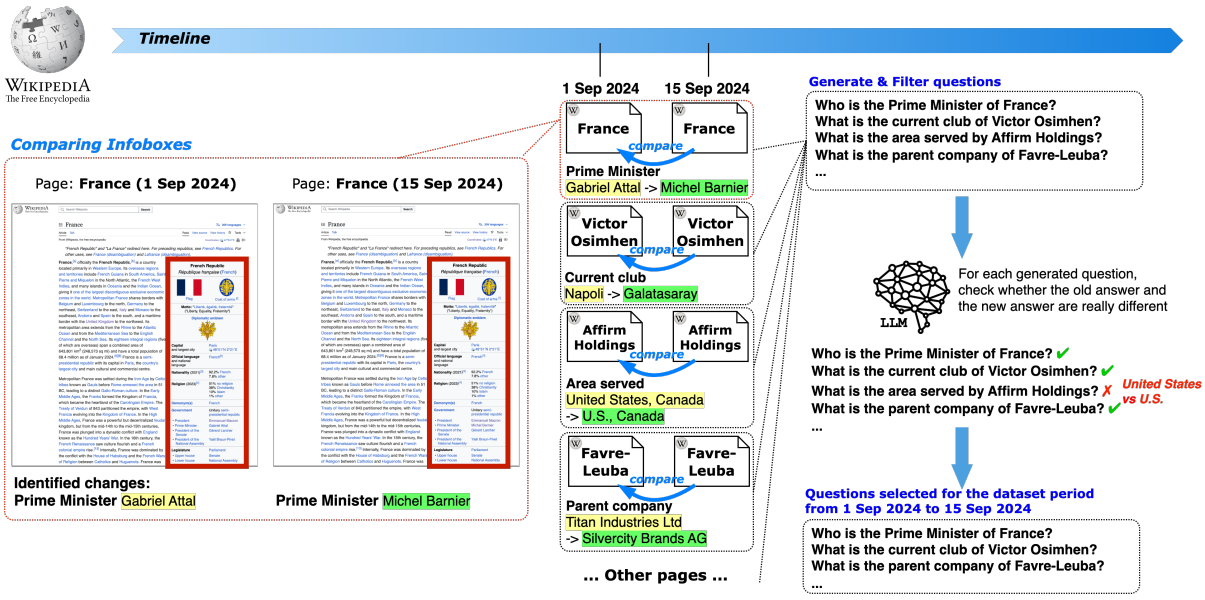

Figure 1: Illustration of the construction process of our DynaQuest benchmark.

to provide a practical testbed for evaluating the ability of LLMs to handle evolving information.

2.2 Data Collection

The data collection involves retrieving Wikipedia pages containing specific types of Infoboxes (e.g., "country", "person") using Wikipedia API.

To identify relevant Infobox types, we analyze the entries listed on the "*List of Infoboxes*" page from Wikipedia. Our selection criteria focuses on Infobox types that 1) are associated with a large number of Wikipedia pages and 2) exhibit meaningful and frequent changes in their attributes. This selection process results in a total of 88 types of Wikipedia Infoboxes, which are further organized into different domains, detailed in Appendix A.1.

After determining the Infobox types, we retrieve Wikipedia pages under the selected types and identify changes in associated attributes and values. We obtain approximately 2.33 million English Wikipedia pages. To address the overwhelming presence of questions from "Sportsperson" domain, we only select 30,000 most frequently updated pages in this domain since 1 September 2023. This refinement finally resulted in approximately 1.98 million Wikipedia pages for dataset collection.

2.3 Benchmark Construction

Evolving Fact Identification. To construct a time-sensitive QA benchmark, we need to identify dynamically evolving facts within a pre-defined time frame. Thus, we first specify a time frame consisting of two dates and then retrieve the corresponding versions of a given Wikipedia page. Next, we compare the Infoboxes from the two versions to identify the fact changes in attributes and values. For example, as shown in Figure 1, by comparing the Wikipedia page "*France*" from 15 September 2024 and 1 September 2024, we identify a knowledge change in the attribute *Prime Minister*, where the old value *Gabriel Attal* was replaced by a new value *Michel Barnier*, corresponding to the real-world event in which French President Emmanuel Macron appointed Michel Barnier as the Prime Minister of France on 5 September 2024, succeeding Gabriel Attal.

Question Generation and Filtering. With identified changes in attributes and values, we generate questions reflecting the involved world knowledge in the Infobox changes. Specifically, we manually design the question templates for each type of attribute in the Infobox and synthesize questions by filling in corresponding attributes. The templates are shown in Appendix A.2. To ensure the quality of generated questions, we apply a filtering process to filter out the undesired questions with patterns specified in Appendix A.3. Additionally, we observe that some identified changes result from minor refinements to the older version. These changes include typo corrections (e.g., Szentlőrinci SE → Szentlőrinc SE), equivalent entity names (e.g., Bucharest, Romania → Bucharest, Kingdom of Romania), abbreviation changes (e.g., United States → U.S.) and equivalent names in different

languages (e.g., Antonio Busce → Antonio Buscè). To avoid incorporating these false positive changes, we leverage a powerful LLM (e.g., Llama-3.1-70B-Instruct) to assess whether each detected difference represents a genuine change. The prompt employed in this verification process is shown in Table 5.

2.4 Benchmark Analysis and Evaluation

The generated question-answer pairs are categorized into three types: (1) *Outdated*, where the answer is derived from the older version of Wikipedia pages; (2) *Updated*, where the answer is based on the newer version of Wikipedia pages, representing updated knowledge and an updated answer; and (3) *New*, where the attribute and corresponding answer do not exist in the older page and are newly added in the newer page.

In our dataset, we include question-answer pairs from the *Updated* and *New* categories within a specified time frame to capture real-world knowledge changes occurring over short periods. Currently, we conduct the benchmark maintenance and update semi-monthly by comparing Wikipedia pages for two time periods: from the 1st to the 15th day of the month, and from the 16th to the last day of the month. Our dataset collection is actively ongoing. In this work, we present the statistics of a snapshot of the benchmark, collected between 16 August 2024 and 15 October 2024, in Table 1. We also present the distribution of questions across domains in Figure 2.

For evaluation on our benchmark, we use *Exact Match (EM)* and *F1 score* as metrics, as the answers in DynaQuest are typically short and concise.

The resources and time required for dataset construction are detailed in Appendix A.5.

	0816-0831	0901-0915	0916-0930	1001-1015
Updated	4440	3126	2113	2764
New	3107	2486	2409	2141
Total	7547	5612	4522	4905

Table 1: The statistics of datasets collected between 16 August 2024 and 15 October 2024.

3 Methodology

3.1 Preliminaries

In this paper, we mainly focus on time-sensitive question answering (Chen et al., 2021; Kim et al., 2024), where the answer might evolve with respect to the time. Retrieval-augmented generation (RAG) has become a critical technique to tackle this problem with supplemented knowledge, where the input to the model $\mathcal{M}_\theta$ parameterized by θ includes a prompt p, question q, and context c:

$$y \sim \mathcal{M}_\theta(y|p,q,c) = \prod_{t=1}^{T} \mathcal{M}_\theta(y_t|p,q,c,y_{<t}). \quad (1)$$

where y represents the generated answer, and y_t denotes the t-th token in the output. In this work, the context c refers to the concatenation of retrieved pieces of text, which can be either the full text of a web page or a segment of texts, e.g., snippet.

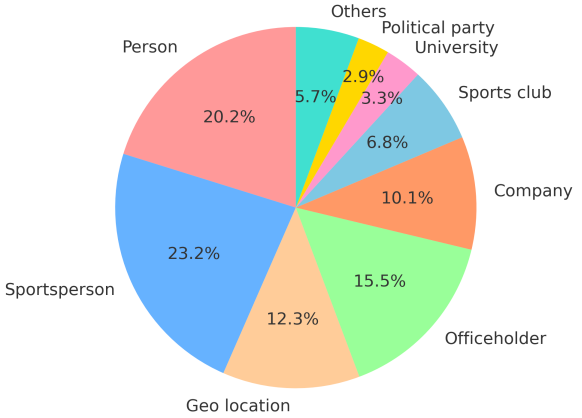

Figure 2: The distribution of questions across domains. The percentages are averaged over the four datasets collected between 16 August 2024 and 15 October 2024.

3.2 Improving Knowledge Integration

LLMs learn knowledge from pre-training, but their parametric knowledge often becomes outdated due to fixed cut-off dates in pre-training corpora. This leads to challenges for tasks requiring up-to-date information in dynamic, time-sensitive domains. A *knowledge conflict* can arise when the knowledge retrieved from external sources contrasts with the LLM's internal knowledge (Zhou et al., 2023; Xie et al., 2024). In RAG, documents are retrieved from external, up-to-date knowledge sources, such as the Internet, to provide the most current and comprehensive context to the LLM. This introduces potential knowledge conflict between the LLM's outdated parametric knowledge and the time-sensitive information in the retrieved context.

To address this challenge, improving knowledge integration with retrieved knowledge is important. Some studies (Zhou et al., 2023; Zhang et al., 2024) demonstrated that it is feasible to enhance the inherent capability of LLMs to flexibly leverage the contextual, parametric knowledge, or both of them

to answer questions in cases of knowledge conflict. In this work, we propose to train the LLM to identify the useful information from the retrieved knowledge in time-sensitive QA while achieving a good balance between retrieved and parametric knowledge to guide the model to selectively incorporate updated information.

3.3 Context-Aware Reinforcement Learning

To better align the model with time-sensitive knowledge in retrieval-augmented generation, we propose Context-Aware Reinforcement Learning (CARL), a learning approach that guides the LLM in identifying and utilizing relevant information from the retrieved context while regulating its reliance on parametric knowledge. The core idea is to construct *pseudo time-sensitive QA data* and introduce a simple yet effective reinforcement learning algorithm, *time-sensitive alignment tuning*, to optimize the model's integration of retrieved and parametric knowledge through a reward mechanism.

Pseudo Time-Sensitive Data. For existing time-sensitive QA data, the questions and answers are derived from earlier versions of Wikipedia, where the knowledge is likely already encoded in the LLM's pre-trained parameters. Training the LLM on such data risks diminishing its effectiveness, as the model may generate answers directly from its internal knowledge rather than referencing the provided context. To address this issue, we propose to construct pseudo time-sensitive QA data where the involved knowledge has not previously been encountered by LLMs. Specifically, given a QA example from existing RAG datasets where an entry consists of a question q, original retrieved context c_o with retrieved documents $\{d_1, ..., d_k\}$, and an original answer a_o, we define a knowledge modification process which modifies the retrieved context c_o and answer a_o to a pseudo context c_m and corresponding pseudo answer a_m while keeping the question q unchanged.

We introduce four types of operations to transform existing data $(q, c_o, a_o) \rightarrow (q, c_m, a_m)$ in order to align the model with time-sensitive knowledge. We leverage the training data from TimeQA (Chen et al., 2021) to construct pseudo QA data in the following four cases. This approach creates explicit training data that is not covered by the model's parametric knowledge. To facilitate understanding of our creation process, we present illustrative examples in Table 13 and Table 14.

• **Case 1**: For a time-sensitive question q, we assign an irrelevant entity to a_m that is unrelated to q or a_o but with the same relation type. We then replace all occurrences of a_o in c_o with a_m to form the modified context c_m.

• **Case 2**: For a time-sensitive question q associated with an answer a_o and corresponding time frame t, we identify another question q' with the same subject as q but involves different answer a' and time frame t'. We remove sentences containing a' to ensure that the answer a' is *not present* in the original context c_o. Then we assign a' to a_m and replace all occurrences of a_o in c_o with a_m to create c_m. This manner will create pseudo context c_m only containing pseudo answer a_m at time frame t but without the original answer a_o.

• **Case 3**: For a time-sensitive question q associated with an answer a_o and corresponding time frame t, we identify another question q' with the same subject as q but involves different answer a' and time frame t'. We ensure that the answer a' is *present* in the original context c_o. Then we assign a' to a_m and swap all occurrences of a_o and a_m in c_o to form c_m. This way will create pseudo context c_m containing both a_o and a_m but they are in different occurrences.

• **Case 4**: Considering the scenario where the answer is not present in the context, for a time-sensitive question q associated with an answer a_o and corresponding time frame t, we remove sentences containing a_o from the original context c_o to form the new context c_m where the answer a_o is not present and keep the answer a_m as a_o. This way will prevent the model bias towards the retrieval information when the context does not contain answer and keep the reliance on the parametric knowledge of LLMs to answer the question.

Cases 1–3 target different types of conflicts between the model's parametric knowledge and retrieved context, while Case 4 includes the case where the answer is absent from retrieval and the model must rely on parametric knowledge. These cases are based on common patterns observed during retrieval. During training data generation, we ensure that only one of the four cases is generated for each original question, avoiding any conflict from repeated questions with differing contexts and answers.

Time-Sensitive Alignment Tuning. Based on the created pseudo QA data, we propose time-sensitive alignment tuning by using reinforcement learning

(RL) to align the model's behavior to generate answers based on retrieved and time-sensitive context. Aligning LLMs through RL has the advantage of preserving the language style of the original pre-trained models (Zhang et al., 2024), whereas supervised fine-tuning may alter the style based on the training data. Specifically, given the pseudo data (q, c, a)[2], we design a deterministic reward function $r(y, a, c)$ through comparing the model prediction y, the gold answer a, and the context c as follows:

$$r(y,a,c) = \begin{cases} +\alpha_0, & \text{if } y = a \ , \\ +\alpha_1, & \text{if } (y \neq a) \wedge \big((y \subseteq a) \vee (a \subseteq y)\big) \ , \\ +\alpha_2, & \text{if } (a \in c) \wedge (y \in c) \wedge (y \not\subseteq a) \\ & \wedge (a \not\subseteq y), \\ -\alpha_3, & \text{if } (a \in c) \wedge (y \notin c), \\ -\alpha_4, & \text{if } (a \notin c) \wedge (y \in c), \\ 0, & \text{Otherwise} \end{cases} \tag{2}$$

where α_k are positive scalars, $y \subseteq a$ indicates that y is a substring of a, and $y \in c$ denotes that y appears as an exact span in c. Based on the reward function, we employ the Proximal Policy Optimization (PPO) algorithm to fine-tune the model (Schulman et al., 2017). The reward design accounts for both scenarios in which the answer is either present or absent in the retrieved context. Heuristically, the model receives a positive reward ($+\alpha_0$ or $+\alpha_1$) when it generates a correct or partially correct answer. When the answer is present in the context, the model is encouraged ($+\alpha_2$) to generate responses using the retrieved knowledge while being penalized ($-\alpha_3$) for relying on outdated parametric knowledge. Conversely, when the answer is absent, the model is penalized ($-\alpha_4$) for generating responses based on the retrieved context, as this may lead to the hallucination problem. With this approach, the model learns to identify and utilize the most relevant and up-to-date information from external sources during question answering, rather than relying preferably on or altering its pre-trained internal knowledge that might be outdated.

4 Experiments

4.1 Experimental Settings

We conduct experiments using three retrieval strategies: without retrieval, with top-5 passages retrieved by a dense retriever based on Google Search results and Wikipedia pages, and with top-5 full texts retrieved from Google Search results and Wikipedia pages. The details of retrieval are provided in Appendix A.6.

In CARL training, for rewards in Eq. 2, we set the values of α_0, α_1, α_2, α_3 and α_4 to 5.0, 4.0, 2.5, 2.5 and 2.5, respectively. The reward values are selected based on a hyperparameter search guided by the training reward graph. We adopt the Kullback-Leibler (KL) divergence between the training policy and the reference policy in regularization and set the penalty coefficient to 0.2. We implement CARL using the TRL (von Werra et al., 2022) library and LoRA (Hu et al., 2022) for parameter-efficient training. We conduct experiments of CARL based on Llama 3.1 8B Instruct, and we denote our trained model as **CARL 8B**. For comparison, we train Llama3.1-8B-Instruct with supervised fine-tuning (SFT) using LoRA and denote the model as **SFT 8B**. For both CARL and SFT, training is conducted using the General prompt template (detailed in Section 4.2).

Moreover, we perform additional evaluation on our datasets using two different evaluation metrics, Answer Rate and F1-Recall[3] introduced in Xu et al. (2025).

4.2 Baselines

We compare our approach to baselines including close-source GPT-4o (OpenAI API checkpoint `gpt-4o-2024-08-06`) (OpenAI, 2024), open-source Llama 3.1 (Llama Team, 2024), state-of-the-art RAG model Self-RAG (Asai et al., 2024), and RiLM (Ko et al., 2024). These approaches are evaluated with the following five types of settings and prompts:

• **No Retrieval.** We use the prompt template in Table 6 to instruct the LLM to generate answers solely based on the question.

• **Retrieval with General Prompt.** We use the general prompt template in Table 7. The LLM is provided with a concatenation of retrieved full texts or passages as the context.

• **Retrieval with Extractive Prompt.** We use the extractive prompt template in Table 8 to instruct the LLM to extract the answer from a concatenation of retrieved full texts or passages.

[2]We omit the subscript m for simplicity.

[3]The Answer Rate (AR) represents the percentage of effective responses, excluding refusal responses, out of all questions, while the F1-Recall (F1-R) is the F1 score computed on effective responses.

	Prompt	0816-0831 EM	0816-0831 F1	0901-0915 EM	0901-0915 F1	0916-0930 EM	0916-0930 F1	1001-1015 EM	1001-1015 F1
Without retrieval									
Llama 3.1 8B Instruct	No Retrieval	4.8	11.7	4.0	10.2	4.6	11.1	4.5	11.4
GPT-4o	No Retrieval	8.9	20.1	11.4	20.8	10.4	19.3	10.4	18.0
With top-5 passages retrieved from Google Search results and Wikipedia page									
Self-RAG 13B	Self-RAG	38.6	47.5	38.5	46.7	39.9	48.5	38.5	46.5
RiLM	RiLM	32.3	47.2	34.7	49.4	35.3	49.1	35.9	49.2
Llama 3.1 8B Instruct	CoT	29.1	41.2	32.0	44.0	33.8	45.4	31.2	42.9
Llama 3.1 8B Instruct	Extractive	37.8	50.7	39.5	52.4	40.0	54.0	40.4	53.1
SFT 8B	Extractive	39.1	51.5	41.1	53.4	41.9	54.2	42.0	53.3
CARL 8B	Extractive	39.9	51.9	41.5	53.2	42.4	54.5	42.3	53.5
Llama 3.1 8B Instruct	General	37.3	50.2	39.2	52.8	39.3	53.3	40.2	52.7
SFT 8B	General	38.2	51.4	39.6	53.2	40.5	54.2	41.1	53.6
CARL 8B (Ours)	General	38.7	51.2	40.9	53.7	41.5	54.5	41.5	53.4
With top-5 full texts retrieved from Google Search results and Wikipedia page									
RiLM	RiLM	47.1	56.6	50.9	59.0	50.5	59.6	50.7	59.6
Llama 3.1 8B Instruct	CoT	36.0	46.2	41.9	52.4	41.3	52.1	39.0	49.3
Llama 3.1 8B Instruct	Extractive	47.6	58.8	49.2	60.1	49.4	60.3	50.0	60.0
SFT 8B	Extractive	47.7	59.1	51.5	62.5	49.5	60.5	51.0	61.3
CARL 8B	Extractive	48.7	**60.2**	52.5	**64.5**	50.4	62.8	52.8	**63.7**
Llama 3.1 8B Instruct	General	47.5	58.9	51.8	60.3	52.0	60.9	52.4	60.7
SFT 8B	General	50.3	59.2	54.6	62.7	53.0	61.9	54.0	62.1
CARL 8B (Ours)	General	**51.4**	59.7	**56.2**	63.8	**54.9**	**63.1**	**55.5**	62.9

Table 2: Performance on collected datasets. The column headers indicate dataset collection periods in 2024.

• **Retrieval with Chain-of-Thought (CoT) Prompt.** The prompt template is shown in Table 9. The LLM is provided with a concatenation of retrieved full texts or passages as the context, and guided to generate reasoning steps before providing the final answer.

• **Self-RAG** (Asai et al., 2024) uses special reflection tokens to improve factuality by evaluating retrieval needs, relevance, and supportiveness of documents. It generates answers per retrieved document, selecting the final answer based on inference scores from these tokens. We use prompt template in Table 10. Self-RAG 13B was built on Llama-2-13B (Touvron et al., 2023b), and we conducted Self-RAG experiments using passages due to the input token limit.

• **RiLM** (Ko et al., 2024) is a framework which reviews its own generated answers, generates feedback based on the retrieved context, and produces improved responses. We adopt the Llama 3.1 8B version released by Ko et al. (2024) and use prompt template in Table 11.

4.3 Main Results and Analysis

Table 2 presents the results of our models and baselines across four versions of our DynaQuest datasets collected from different time periods.

In the experiments without retrieval, GPT-4o consistently achieves the best performance among the evaluated models, with EM scores ranging from 8.9 to 11.4 and F1 scores from 18.0 to 20.8. The overall performance across all these state-of-the-art LLMs is low, exhibiting the significant challenge posed by knowledge updates in our constructed datasets for time-sensitive QA. When incorporating top-5 passages retrieved by Contriever, Llama 3.1 8B Instruct exhibit notable performance improvements compared to the Without Retrieval setting. Without RAG training, Llama 3.1 8B Instruct performs on par with Self-RAG on our datasets. With both General and Extractive prompts, CARL 8B outperforms its base model Llama 3.1 8B Instruct.

The CoT prompt shows limited effectiveness in leveraging multiple full texts, underperforming compared to all other prompt settings. Excessive reasoning can introduce unsupported inferences, increasing the risk of hallucinations in dynamic QA tasks. This aligns with the observed performance of CoT in Xu et al. (2025).

Recent LLMs like Llama 3.1, with context windows over 100 thousand tokens, enable full text to be a more effective RAG configuration. Across all

four datasets, all Llama 3.1 pre-trained models and CARL 8B consistently achieve performance gains over the passages setting using the same prompt. CARL 8B demonstrates further improvements with both General and Extractive prompts, outperforming Llama 3.1 8B Instruct and SFT 8B. We performed a paired bootstrap significance test on the F1 scores of CARL and SFT outputs, and the improvement achieved by CARL is statistically significant ($p < 0.05$). More results on further periods can be observed in the additional results shown in Table 15.

The evaluation results for Answer Rate and F1-Recall are presented in Table 16 and Table 17. With an overall Answer Rate exceeding 93%, the F1-Recall comparison aligns with the F1 results in Table 2 and 15. CARL 8B achieves the best performance across all datasets.

	0816–0831		0901–0915	
	EM	F1	EM	F1
CARL	**51.4**	**59.7**	**56.2**	**63.8**
w/o Case 1	49.4	59.3	53.9	62.0
w/o Case 2	49.5	59.2	54.0	62.1
w/o Case 3	49.8	59.4	54.3	62.4
w/o Case 4	50.2	59.4	54.5	62.5
	0916–0930		1001–1015	
	EM	F1	EM	F1
CARL	**54.9**	**63.1**	**55.5**	**62.9**
w/o Case 1	53.4	61.9	53.8	61.8
w/o Case 2	53.5	62.1	54.2	62.0
w/o Case 3	53.9	62.2	54.1	62.2
w/o Case 4	53.8	62.3	54.3	62.1

Table 3: Ablation study on different training case types. Results are based on the full-text setting with the General prompt.

	PopQA	TriviaQA
ChatGPT*	50.8	65.7
GPT-4o	52.0	**75.1**
Self-RAG 13B*	**55.8**	69.3
Llama 3.1 8B Instruct	53.5	68.9
SFT 8B	53.6	70.8
CARL 8B (Ours)	<u>55.6</u>	<u>72.9</u>

Table 4: Results of RAG experiments on additional datasets. The best and second-best results are bolded and underlined respectively. * results from Asai et al. (2024).

4.4 Further Analysis

We present ablation study on different training case types in Table 3. Removing any single case from CARL leads to a consistent drop in EM/F1 across all time windows, confirming that each case contributes to performance. Case 1 shows the largest drop when removed, indicating its importance in alignment between parametric and retrieved knowledge. Case 2 and Case 3 help the model adapt to conflicting information, and their removal reduces robustness. Case 4 addresses reliance on outdated parametric knowledge and improves model when no correct answer is retrieved.

To evaluate the generalization capability of CARL, we conduct experiments on QA datasets where some answers are not present in the retrieved context. Specifically, we adopt the evaluation of two short-form open-domain QA datasets used in Asai et al. (2024), including a PopQA long-tail subset (Mallen et al., 2023) and a TriviaQA test set (Joshi et al., 2017). Following the RAG experimental setup, we utilize the top-5 documents and the evaluation metric of Accuracy provided in Asai et al. (2024).

The experimental results are presented in Table 4. CARL 8B demonstrates improved performance on both datasets compared to its base model, Llama 3.1 8B Instruct. In PopQA experiments, CARL 8B achieves performance comparable to Self-RAG 13B. In TriviaQA experiments, CARL 8B outperforms Self-RAG and ChatGPT by a noticeable margin, while only remaining behind GPT-4o. These results suggest that the improved knowledge integration capability of CARL enhances its generalization to QA scenarios where the correct answer is not present in the retrieved context.

5 Related Work

Time-sensitive Question Answering. Recent works on time-related QA datasets adopt diverse approaches to address temporal reasoning and knowledge-based queries. TempReason (Tan et al., 2023) and Complex-TR (Tan et al., 2024) focus on temporal reasoning with questions derived from a static Wikidata corpus. Mousavi et al. (2024) proposed DyKnow framework to evaluate model completions of time-sensitive facts using Wikidata. In contrast, our dataset continuously updates question-answer pairs, providing a dynamic QA benchmark for evolving knowledge. RealtimeQA (Kasai et al., 2023) focuses on time-sensitive QA with questions and answers curated from news quizzes on news websites. FreshQA (Vu et al., 2023) includes natural questions with answers regularly updated by human annotators. CDQA (Xu et al., 2025) consists of dynamic Chinese questions from news

media, annotated by humans. Our dataset introduces an automated collection method, covering diverse domains with key updates derived from Wikipedia Infoboxes. GrowOver (Ko et al., 2024) and EvolvingQA (Kim et al., 2024) rely on full-text comparisons of Wikipedia versions to generate QA pairs but encounter incompleteness and difficulty in identifying precise changes. Our dataset leverages Wikipedia Infoboxes for clearer and more accurate real-world updates.

Retrieval-Augmented Generation (RAG). LLMs, reliant on pre-training data, often lack up-to-date knowledge. RAG mitigates this by retrieving relevant documents and incorporating them into model inputs, as shown in Lewis et al. (2020) and Borgeaud et al. (2022). With the increasing context window size of recent LLMs (Llama Team, 2024; OpenAI, 2024), RAG has become more effective by allowing substantial retrieved information to be included in the input. Fine-tuning retrievers and readers jointly has been shown to align retrieval and generation, as explored in works like ATLAS (Izacard et al., 2023), Promptgator (Dai et al., 2023), and Shao et al. (2023). Recently, Self-RAG (Asai et al., 2024) introduced a self-reflective mechanism that uses critique tokens and enables models to evaluate the relevance and utility of retrieved information, enhancing the integration of retrieved knowledge into the generation process. RetRobust (Yoran et al., 2024) is designed to prompt LLMs to assess document relevance before answering, enhancing robustness to irrelevant context. Research has also explored improved decoding strategies, such as iterative retrieval (Trivedi et al., 2023; Shao et al., 2023), contrastive decoding (Shi et al., 2024), and dynamic context refinement (Jiang et al., 2023) to improve the generation quality. Toolformer (Schick et al., 2023) incorporates tools during generation for knowledge-intensive tasks, further enhancing the effectiveness of retrieval-augmented systems.

6 Conclusion

In this paper, we introduce DynaQuest, a dynamic question answering dataset reflecting real-world knowledge updates. Built using Wikipedia Infoboxes, DynaQuest offers an evolving benchmark for evaluating LLMs on time-sensitive QA tasks. To address the challenges posed by dynamic knowledge updates in DynaQuest, we propose CARL, a context-aware reinforcement learning method that improves LLM integration with retrieved, up-to-date knowledge. Experimental results across multiple dataset periods demonstrated the effectiveness of our approach against strong baselines.

Limitations

Our dataset relies on automated pipelines to generate question-answer pairs from Wikipedia Infoboxes. Due to the flexible editing nature of Wikipedia, real-world changes may occasionally experience delays in being reflected, and minor errors in validating these changes can occur. While proposed CARL effectively improves integration with retrieved knowledge, its reliance on reinforcement learning introduces computational overhead, which could be further optimized in future research.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *ICLR*.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, and 9 others. 2022. Improving language models by retrieving from trillions of tokens. In *ICML*, pages 2206–2240.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *NeurIPS*.

Wenhu Chen, Xinyi Wang, William Yang Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *NeurIPS Datasets and Benchmarks*.

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot dense retrieval from 8 examples. In *ICLR*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *TACL*, 10:257–273.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *ICLR*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *TMLR*.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot learning with retrieval augmented language models. *JMLR*, 24(251):1–43.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *EMNLP*, pages 7969–7992.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, pages 1601–1611.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Velocity Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2023. Realtime qa: what's the answer right now? In *NeurIPS Datasets and Benchmarks*.

Yujin Kim, Jaehong Yoon, Seonghyeon Ye, Sangmin Bae, Namgyu Ho, Sung Ju Hwang, and Se-Young Yun. 2024. Carpe diem: On the evaluation of world knowledge in lifelong language models. In *NAACL-HLT*, pages 5401–5415.

Dayoon Ko, Jinyoung Kim, Hahyeon Choi, and Gunhee Kim. 2024. GrowOVER: How can LLMs adapt to growing real-world knowledge? In *ACL*, pages 3282–3308.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NeurIPS*, pages 9459–9474.

Llama Team. 2024. The llama 3 herd of models. *arXiv preprint*, arXiv:2407.21783.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *ACL*, pages 9802–9822.

Seyed Mahed Mousavi, Simone Alghisi, and Giuseppe Riccardi. 2024. DyKnow: Dynamically verifying time-sensitive factual knowledge in LLMs. In *Findings of EMNLP*, pages 8014–8029.

OpenAI. 2024. Gpt-4 technical report. *arXiv preprint*, arXiv:2303.08774.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Technical Report*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint*, arXiv:2302.04761.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint*, arXiv:1707.06347.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of EMNLP*, pages 9248–9274.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024. Trusting your evidence: Hallucinate less with context-aware decoding. In *NAACL-HLT*, pages 783–791.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models. In *ACL*, pages 14820–14835.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2024. Towards robust temporal reasoning of large language models via a multi-hop QA dataset and pseudo-instruction tuning. In *Findings of ACL*, pages 6272–6286.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint*, arXiv:2307.09288.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *ACL*, pages 10014–10037.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2022. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint*, arXiv:2310.03214.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2024. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *ICLR*.

Zhikun Xu, Yinghui Li, Ruixue Ding, Xinyu Wang, Boli Chen, Yong Jiang, Haitao Zheng, Wenlian Lu, Pengjun Xie, and Fei Huang. 2025. Let LLMs take on the latest challenges! a Chinese dynamic question answering benchmark. In *COLING*, pages 10435–10448.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *ICLR*.

Zongmeng Zhang, Yufeng Shi, Jinhua Zhu, Wengang Zhou, Xiang Qi, Peng Zhang, and Houqiang Li. 2024. Trustworthy alignment of retrieval-augmented large language models via reinforcement learning. In *ICML*, pages 59827–59850.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of EMNLP*, pages 14544–14556.

A Appendix

A.1 Infobox Types

We present the Infobox types used in our data collection process, organized into different domains detailed below.

Sportsperson: AFL biography, athlete, badminton player, baseball biography, basketball biography, boxer, Canadian Football League biography, CBB biography, CFL biography, CFL player, chess player, college basketball, college coach, college football player, cyclist, darts player, F1 driver, field hockey player, figure skater, football biography, golfer, gridiron football person, gymnast, handball biography, ice hockey biography, ice hockey player, judoka, martial artist, motorcycle rider, NFL biography, NFL player, NPB player, professional wrestler, rugby biography, rugby league biography, rugby union biography, skier, snooker player, Speedway rider, sportsperson, swimmer, table tennis player, tennis biography, volleyball biography, volleyball player

Person: academic, actor, architect, artist, astronaut, comedian, comics creator, criminal, economist, medical person, military person, model, musical artist, pageant titleholder, person, philosopher, police officer, scientist, serial killer, Twitch streamer, writer, YouTube personality

Officeholder: judge, officeholder, politician

Company: airline, company

Sports club: basketball club, football club

Political party: political party

Geographical location: country, islands, settlement

University: university

Others: award, bishop styles, Christian leader, journal, monarch, noble, religious biography, royalty, saint

A.2 Templates for Question Generation

Templates for question generation and corresponding examples are shown in Table 12. Here, {page} refers to the title of the page, {ARG1} denotes an attribute, and {ARG2} represents additional information associated with {ARG1}, as retrieved from the Wikipedia API. For instance, in the example of "Who won Golden Pen Award at Writers and Illustrators of the Future?", the retrieved {ARG1} is "Award 1 Name" and {ARG2} specifies the award name "Golden Pen Award", only {ARG2} is included in the question. In the example of "What is the Gini index of Syria in 2022?", the retrieved {ARG1} is "Gini index" and {ARG2} specifies the year "2022", both {ARG1} and {ARG2} are included in the question.

A.3 Undesired Questions

Undesired questions are those where the answers are either not expected to change, involve numbers that are not fixed, or are overly open-ended, making them difficult to evaluate. Such questions are filtered out from the dataset, including the following patterns:

> What is the birth date of ... ?
> What is the death date of ... ?
> Where is the birth place of ... ?
> Where is the death place of ... ?
> What is ... known for?
> What is the population of ... ?
> What is the capacity of home stadium of ... ?

A.4 Resources

Wikipedia API: https://www.mediawiki.org/wiki/API:Main_page
Wikipedia "*List of Infoboxes*" page: https://en.wikipedia.org/wiki/Wikipedia:List_of_infoboxes
Google Custom Search: https://developers.google.com/custom-search/v1/overview
Contriever-MSMARCO: https://huggingface.co/facebook/contriever-msmarco
GPT-4o API: https://platform.openai.com/docs/models

Llama 3.1 8B Instruct: https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

A.5 Cost of Dataset Construction

The dataset construction pipeline consists of Wikipedia data collection, question generation, and question filtering. In each dataset construction cycle, Wikipedia data collection takes about 6 hours, primarily due to the Wikipedia API rate limit. Question generation, based on Infobox comparisons, takes less than 5 minutes and requires minimal computational resources. The question filtering process, which uses Llama-3.1-70B-Instruct, takes around 20 minutes with two H100 80GB GPUs. The pipeline does not require human annotation, enabling timely and scalable updates.

A.6 Retrieval Setup

Google Search is a widely used and intuitive knowledge source, making it a realistic scenario for real-world knowledge retrieval. To ensure that retrieved information reflects real-time knowledge, we retrieve relevant web pages by querying the generated question on Google Search engine and collecting the search results on the same day as dataset construction. Search results are obtained using the Google Custom Search API. The top 5 web pages are selected as the initial retrieval set. If the ground-truth Wikipedia page corresponding to the question is not included among the top 5 pages from Google Search results, we retain the top 4 pages and additionally include the ground-truth Wikipedia page to ensure the presence of the expected knowledge, effectively simulating the gold retrieval setting.

Based on this retrieval setup, we use the 5 selected web pages as the knowledge source for each question. Following Asai et al. (2024), for experiments involving passage-level retrieval, the full text of each page is segmented into 100-word passages, which are then ranked using Contriever (Izacard et al., 2022) (MSMARCO version). The top 5 passages are selected for corresponding experiments. For over 78.8% of the questions, the ground-truth passage containing the correct answer is included in the top 5 results by Contriever.

A.7 Implementation Details

The size of training data generated from TimeQA is 10,000. We train CARL with PPO using the TRL library. LoRA is applied with a rank of 64 and targeting modules `{q_proj,v_proj,k_proj,o_proj}`. The batch size is set to 2, and the learning rate is configured as 2×10^{-7}. The training epoch is set to 2. During SFT model training, LoRA is applied with a rank of 64 and target modules `{q_proj,v_proj,k_proj,o_proj}`. The batch size is set to 24, and the learning rate is configured as 2×10^{-5}. The training epoch is set to 2. The training of each model was performed once on a Nvidia H100 GPU. The training times of CARL and SFT are 12 and 8 hours, respectively.

A.8 Dataset Licensing and Availability

Wikipedia is licensed under CC BY-SA 4.0. The DynaQuest datasets will be released under the same CC BY-SA 4.0 license.

The datasets used in this paper are collected semi-monthly, and we plan to continue semi-monthly updates in the future. We will provide QA pairs along with Google Search results and Wikipedia pages for the dataset used in this paper.

Instruction: Given the context of the question and the provided texts as answers, verify whether Text B is genuinely different from Text A for the specified question. Take into account cases such as typographical corrections, place name equivalence, abbreviations, or name variations across languages. Answer only "Yes" if Text B is genuinely different from Text A, or "No" if they represent the same entity. Page: {page_name} Question: {question} Text A: {old_answer} Text B: {new_answer}

Table 5: Prompt template for verifying the generated question-answer pairs.

Provide the shortest answer to the question without any introductory phrases or explanations: {question}

Table 6: Prompt template of **No Retrieval**.

Additional information: {context} Provide the shortest answer to the question without any introductory phrases or explanations: {question}

Table 7: Prompt template of **General**.

Context: {context} Question: {question} Instruction: Extract the shortest possible answer directly from the context. The answer must be taken exactly as it appears in the context, without any modifications, rewording, or additions. Do not infer or generate content beyond what is explicitly stated. If multiple possible answers exist, choose the shortest one that fully answers the question.

Table 8: Prompt template of **Extractive**.

Context: {context} Question: {question} Based on the provided context and your knowledge, please reason through the question step by step. First, provide a clear explanation of your reasoning process. Then, provide the shortest answer as the final answer at the end in the following format: "The final answer is ..."

Table 9: Prompt template of **Chain-of-Thought (CoT)** prompting.

Instruction: Provide the shortest response without any introductory phrases or explanations to answer the following question. ## Input: {question} ### Response: [Retrieval]<paragraph> {context} </paragraph>

Table 10: Prompt template of **Self-RAG**. Adapted from Asai et al. (2024).

Use the following context to answer the question. Context: {context} Question: {question} Answer:

Table 11: Prompt template of **RiLM**. Adapted from Ko et al. (2024)

Template	Example
What is the {ARG1} of {page}?	What is the reign period of Muhammad bin Talal Al Rashid?
What {ARG1} did {page} play?	What position did Jeremiah Pharms Jr. play?
Which {ARG1} did {page} attend?	Which school did Fraser McReight attend?
When is the {ARG1} of {page}?	When is the career end of Michael Carter-Williams?
Which {ARG1} does {page} play in?	Which league does Skawinka Skawina play in?
Which {ARG1} does {page} play for?	Which team does Cole Irvin play for?
What are the {ARG1} of {page}?	What are the active years of Clint Eastwood filmography?
Who is the {ARG1} of {page}?	Who is the manager of Phu Dong Ninh Binh FC?
Who is/are the {ARG1} of {page}?	Who is/are the spouse(s) of Dallas Roberts?
What {ARG1} does {page} play?	What sport does Rick McCrank play?
When did {page} {ARG1}?	When did Florida Express cease operations?
How many {ARG1} has {page} recorded?	How many Moto2 Poles has Fermín Aldeguer recorded?
What is the {ARG1} by {page}?	What is the area served by Tata Motors?
Where is the {ARG1} of {page}?	Where is the HQ location of Olga TV?
What does {page} {ARG1}?	What does Paul Lauterbur known For?
How many {ARG1} does {page} have?	How many members does The Way of Courage have?
What group is {page} a {ARG1}?	What group is Rick Savage a current member of?
What group was {page} a {ARG1}?	What group was El Hefe a past member of?
What is {page} a {ARG1}?	What is June of 44 a spinoff of?
Who are the {ARG1} of {page}?	Who are the parents of Jacinda Ardern?
Who were the {ARG1} of {page}?	Who were the past members of Coriky?
Where are the {ARG1} of {page}?	Where are the headquarters of Libertarian Party of Maine?
{page} is {ARG1} what?	RDCWorld is created by what?
When was {page} {ARG1}?	When was Colorado Rapids founded?
What is the number of {ARG1} caused by {page}?	What is the number of fatalities caused by Katherine Knight?
What is {page} {ARG1}?	What is Jayuya, Puerto Rico named for?
What is the {ARG1} for {page}?	What is the number of locations for Żabka (convenience store)?
What is {page}'s {ARG1} in {ARG2}?	What is Landmark Theatres's revenue in 2023?
How many {ARG1} are served by {page}?	How many destination airports are served by Aurigny?
Where is/are the {ARG1} of {page}?	Where is/are the headquarters of FlyErbil?
What is the {ARG1} of {page} in {ARG2}?	What is the Gini index of Syria in 2022?
Who is the {ARG2} of {page}?	Who is the Additional Deputy Commissioner of Jonai?
What is the {ARG1} of {page} home stadium?	What is the location of Indiana Mad Ants home stadium?
{page} is {ARG1}?	Alden John Bell is consecrated by?
What was the {ARG1} of {page}?	What was the succession of Simeon Saxe-Coburg-Gotha?
When was {page}'s {ARG1}?	When was Otto IV, Holy Roman Emperor's coronation?
When did {page}'s {ARG1}?	When did Rogelio del Rosario Martinez's term start?
Where was {page} {ARG1}?	Where was Vinzenz Eduard Milde buried?
{page} was {ARG1}?	Ana de Jesús was beatified by?
Who is the {ARG1} while {page} is in office?	Who is the governor while Rajyavardhan Singh Rathore is in office?
When did {page}'s {ARG1}?	When did Pridi Banomyong's term start?
In which {ARG1} is {page} awarded?	In which country is Vayalar Award awarded?
Who has {ARG1} of {page}?	Who has most awards/wins of Nandi Award for Best Actor?
What are the {ARG1} by {page}?	What are the prizes awarded by National Film Award for Best Lyrics?
Who won {ARG2} at {page}?	Who won Golden Pen Award at Writers and Illustrators of the Future?

Table 12: Templates for question generation and generated examples.

Original data from TimeQA
Question q: What position did Eleftherios Papageorgopoulos take in Dec 1985?
Context c_o:
Eleftherios Papageorgopoulos ( ) is a Greek politician who served as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet from August to September 2015 . Early life and education . Papageorgopoulos was born in Chalcis on 6 October 1947 . He studied law at the University of Athens .
Papageorgopoulos served as President of Chalcis Municipality on the City Council from 1 January 1983 to 4 March 1985 . From 1 May 1985 to 31 December 1986 , Papageorgopoulos served as Mayor of Chalcis .
Papageorgopoulos was first elected as a Member of the Hellenic Parliament for Euboea in the June 1989 election .
He was subsequently re-elected in November 1989 , 1990 , 1993 , 1996 and 2000 . As an MP , he was a member of the Standing Committee on Economic Affairs and a member of the Commission for the Revision of the Constitution . He served for some time as the parliamentary spokesperson for New Democracy .
On 28 August 2015 , Papageorgopoulos was sworn in as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet , serving until 23 September 2015 .
Papageorgopoulos is married to Amalia Passa and has one daughter and two sons .
Answer a_o: Mayor

[Case 1]
Irrelevant entity selected from TimeQA with the same relation type (position held): Deputy Minister of Irrigation
Question q: What position did Eleftherios Papageorgopoulos take in Dec 1985?
Context c_m:
Eleftherios Papageorgopoulos ( ) is a Greek politician who served as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet from August to September 2015 . Early life and education . Papageorgopoulos was born in Chalcis on 6 October 1947 . He studied law at the University of Athens .
Papageorgopoulos served as President of Chalcis Municipality on the City Council from 1 January 1983 to 4 March 1985 . From 1 May 1985 to 31 December 1986 , Papageorgopoulos served as Deputy Minister of Irrigation of Chalcis .
Papageorgopoulos was first elected as a Member of the Hellenic Parliament for Euboea in the June 1989 election . He was subsequently re-elected in November 1989 , 1990 , 1993 , 1996 and 2000 . As an MP , he was a member of the Standing Committee on Economic Affairs and a member of the Commission for the Revision of the Constitution . He served for some time as the parliamentary spokesperson for New Democracy .
On 28 August 2015 , Papageorgopoulos was sworn in as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet , serving until 23 September 2015 .
Papageorgopoulos is married to Amalia Passa and has one daughter and two sons .
Answer a_m: Deputy Minister of Irrigation

[Case 2]
Another question q' with the same subject but a different time frame: What position did Eleftherios Papageorgopoulos take between Dec 1993 and Jul 1996?
with answer a': Member of the Hellenic Parliament
Question q: What position did Eleftherios Papageorgopoulos take in Dec 1985?
Context c_m:
Eleftherios Papageorgopoulos ( ) is a Greek politician who served as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet from August to September 2015 . Early life and education . Papageorgopoulos was born in Chalcis on 6 October 1947 . He studied law at the University of Athens .
Papageorgopoulos served as President of Chalcis Municipality on the City Council from 1 January 1983 to 4 March 1985 . From 1 May 1985 to 31 December 1986 , Papageorgopoulos served as Member of the Hellenic Parliament of Chalcis .
He was subsequently re-elected in November 1989 , 1990 , 1993 , 1996 and 2000 . As an MP , he was a member of the Standing Committee on Economic Affairs and a member of the Commission for the Revision of the Constitution . He served for some time as the parliamentary spokesperson for New Democracy .
On 28 August 2015 , Papageorgopoulos was sworn in as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet , serving until 23 September 2015 .
Papageorgopoulos is married to Amalia Passa and has one daughter and two sons .
Answer a_m: Member of the Hellenic Parliament
(We remove the sentence "Papageorgopoulos was first elected as a Member of the Hellenic Parliament for Euboea in the June 1989 election ." from the context to avoid conflicting information within the content.)

Table 13: Illustration of training data generation for CARL. Finally we use question q, context c_m and answer a_m from each generation process during training.

[Case 3]
Another question q' with the same subject but a different time frame: What position did Eleftherios Papageorgopoulos take between Dec 1993 and Jul 1996?
with answer a': Member of the Hellenic Parliament
Question q: What position did Eleftherios Papageorgopoulos take in Dec 1985?
Context c_m:
Eleftherios Papageorgopoulos ( ) is a Greek politician who served as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet from August to September 2015 . Early life and education . Papageorgopoulos was born in Chalcis on 6 October 1947 . He studied law at the University of Athens .
Papageorgopoulos served as President of Chalcis Municipality on the City Council from 1 January 1983 to 4 March 1985 .
From 1 May 1985 to 31 December 1986 , Papageorgopoulos served as Member of the Hellenic Parliament of Chalcis .
Papageorgopoulos was first elected as a Mayor for Euboea in the June 1989 election .
He was subsequently re-elected in November 1989 , 1990 , 1993 , 1996 and 2000 . As an MP , he was a member of the Standing Committee on Economic Affairs and a member of the Commission for the Revision of the Constitution . He served for some time as the parliamentary spokesperson for New Democracy .
On 28 August 2015 , Papageorgopoulos was sworn in as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet , serving until 23 September 2015 .
Papageorgopoulos is married to Amalia Passa and has one daughter and two sons .
Answer a_m: Member of the Hellenic Parliament

[Case 4]
The sentence containing the original answer a_o (Mayor) is removed from the original context c_o to generate the modified context c_m. Since the answer is no longer present in context c_m, the model is expected to utilize its parametric knowledge to generate the answer.
Question q: What position did Eleftherios Papageorgopoulos take in Dec 1985?
Context c_m:
Eleftherios Papageorgopoulos ( ) is a Greek politician who served as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet from August to September 2015 . Early life and education . Papageorgopoulos was born in Chalcis on 6 October 1947 . He studied law at the University of Athens .
Papageorgopoulos served as President of Chalcis Municipality on the City Council from 1 January 1983 to 4 March 1985 .
Papageorgopoulos was first elected as a Member of the Hellenic Parliament for Euboea in the June 1989 election .
He was subsequently re-elected in November 1989 , 1990 , 1993 , 1996 and 2000 . As an MP , he was a member of the Standing Committee on Economic Affairs and a member of the Commission for the Revision of the Constitution . He served for some time as the parliamentary spokesperson for New Democracy .
On 28 August 2015 , Papageorgopoulos was sworn in as a Minister of State in Vassiliki Thanou-Christophilous caretaker cabinet , serving until 23 September 2015 .
Papageorgopoulos is married to Amalia Passa and has one daughter and two sons .
Answer a_m: Mayor

Table 14: (Continued) Illustration of training data generation for CARL. Finally we use question q, context c_m and answer a_m from each generation process during training.

	Prompt	1016-1031 EM	1016-1031 F1	1101-1115 EM	1101-1115 F1	1116-1130 EM	1116-1130 F1
Without retrieval							
Llama 3.1 8B Instruct	No Retrieval	2.9	9.5	3.5	10.1	3.0	9.9
GPT-4o	No Retrieval	8.2	18.5	9.1	18.2	9.9	20.6
With top-5 passages retrieved from Google Search results and Wikipedia page							
Self-RAG 13B	Self-RAG	38.0	46.0	37.5	45.1	39.6	47.6
RiLM	RiLM	32.8	46.8	34.5	47.4	34.8	48.6
Llama 3.1 8B Instruct	CoT	29.3	41.2	30.1	41.8	30.6	41.5
Llama 3.1 8B Instruct	Extractive	36.7	49.9	39.4	51.0	39.3	52.0
SFT 8B	Extractive	38.1	50.1	39.6	51.5	40.3	52.7
CARL 8B (Ours)	Extractive	39.1	50.6	41.0	51.7	43.6	53.9
Llama 3.1 8B Instruct	General	36.8	50.1	38.8	50.8	39.0	52.0
SFT 8B	General	37.8	51.0	39.9	51.9	40.3	53.1
CARL 8B (Ours)	General	38.1	50.7	40.1	51.5	41.2	53.3
With top-5 full texts retrieved from Google Search results and Wikipedia page							
RiLM	RiLM	47.0	54.5	48.2	56.0	50.0	57.9
Llama 3.1 8B Instruct	CoT	38.7	49.1	38.2	48.0	38.3	48.5
Llama 3.1 8B Instruct	Extractive	47.4	57.9	47.9	57.5	50.3	60.2
SFT 8B	Extractive	47.8	59.8	48.4	57.7	50.8	61.5
CARL 8B (Ours)	Extractive	48.9	60.7	49.4	60.1	52.0	**63.8**
Llama 3.1 8B Instruct	General	49.5	57.7	49.7	58.0	52.0	60.2
SFT 8B	General	51.6	59.9	51.4	59.5	54.4	62.2
CARL 8B (Ours)	General	**53.3**	**60.9**	**53.3**	**60.7**	**56.6**	63.7

Table 15: Additional results on datasets collected between 16 October 2024 and 30 November 2024.

		0816-0831		0901-0915		0916-0930		1001-1015	
	Prompt	F1-R	AR	F1-R	AR	F1-R	AR	F1-R	AR
Without retrieval									
Llama 3.1 8B Instruct	No Retrieval	11.8	(99.9)	10.2	(100)	11.1	(99.9)	11.4	(99.8)
GPT-4o	No Retrieval	20.1	(99.9)	20.8	(100)	19.4	(99.7)	18.1	(99.9)
With top-5 passages retrieved from Google Search results and Wikipedia page									
Self-RAG 13B	Self-RAG	47.5	(99.9)	46.7	(99.9)	48.5	(99.9)	46.5	(100)
RiLM	RiLM	47.8	(99.6)	49.4	(99.6)	49.5	(99.5)	50.1	(99.6)
Llama 3.1 8B Instruct	CoT	42.2	(96.3)	45.1	(96.2)	46.5	(96.3)	44.1	(96.3)
Llama 3.1 8B Instruct	Extractive	51.1	(95.0)	53.2	(94.0)	55.0	(93.2)	54.3	(93.1)
SFT 8B	Extractive	51.6	(97.6)	53.9	(97.3)	55.3	(97.3)	54.3	(97.0)
CARL 8B (Ours)	Extractive	51.8	(98.7)	53.8	(98.8)	56.1	(98.5)	54.9	(98.5)
Llama 3.1 8B Instruct	General	51.0	(97.2)	53.4	(96.7)	54.4	(95.9)	53.6	(96.2)
SFT 8B	General	51.5	(99.2)	54.0	(99.2)	54.5	(98.9)	54.1	(98.9)
CARL 8B (Ours)	General	51.9	(99.0)	54.1	(98.9)	55.1	(98.7)	54.1	(98.7)
With top-5 full texts retrieved from Google Search results and Wikipedia page									
RiLM	RiLM	59.0	(97.5)	59.7	(97.0)	60.5	(96.3)	61.0	(96.3)
Llama 3.1 8B Instruct	CoT	47.5	(96.8)	53.2	(96.7)	53.8	(96.4)	51.2	(95.9)
Llama 3.1 8B Instruct	Extractive	58.9	(97.2)	60.8	(96.7)	61.0	(96.6)	60.8	(96.3)
SFT 8B	Extractive	59.6	(99.9)	63.5	(99.9)	60.5	(99.8)	61.3	(100)
CARL 8B	Extractive	60.2	(99.9)	**64.5**	(100)	62.8	(99.9)	**63.7**	(100)
Llama 3.1 8B Instruct	General	59.5	(97.9)	61.7	(97.6)	61.9	(97.2)	61.4	(97.3)
SFT 8B	General	59.6	(99.2)	63.2	(99.3)	62.5	(99.1)	62.6	(99.0)
CARL 8B (Ours)	General	**60.3**	(99.0)	64.4	(99.1)	**63.9**	(98.8)	63.6	(98.9)

Table 16: Performance on datasets collected between 16 August 2024 and 15 October 2024, with F1-Recall (F1-R) and Answer Rate (AR, in %) as evaluation metrics.

		1016-1031		1101-1115		1116-1130	
	Prompt	F1-R	AR	F1-R	AR	F1-R	AR
Without retrieval							
Llama 3.1 8B Instruct	No Retrieval	9.6	(99.9)	10.1	(99.9)	9.9	(99.9)
GPT-4o	No Retrieval	18.5	(99.9)	18.2	(99.9)	20.7	(99.7)
With top-5 passages retrieved from Google Search results and Wikipedia page							
Self-RAG 13B	Self-RAG	46.0	(100)	45.1	(100)	47.6	(99.9)
RiLM	RiLM	46.8	(99.1)	47.4	(99.1)	48.6	(99.5)
Qwen2.5 7B Instruct	CoT	42.8	(96.1)	42.4	(96.1)	43.0	(95.8)
Llama 3.1 8B Instruct	CoT	43.0	(95.5)	42.9	(95.8)	42.5	(95.3)
Llama 3.1 8B Instruct	Extractive	50.1	(93.3)	52.2	(93.3)	53.1	(93.7)
SFT 8B	Extractive	50.3	(97.2)	52.3	(97.3)	53.5	(96.9)
CARL 8B (Ours)	Extractive	51.3	(98.4)	52.0	(98.7)	54.3	(98.3)
Llama 3.1 8B Instruct	General	51.0	(96.8)	52.1	(96.7)	52.9	(96.3)
SFT 8B	General	51.5	(99.1)	52.3	(99.1)	53.4	(99.0)
CARL 8B (Ours)	General	51.1	(98.8)	52.1	(98.9)	53.9	(98.7)
With top-5 full texts retrieved from Google Search results and Wikipedia page							
RiLM	RiLM	58.5	(97.0)	57.8	(98.0)	60.8	(96.8)
Llama 3.1 8B Instruct	CoT	50.0	(96.5)	49.3	(96.7)	50.5	(96.2)
Llama 3.1 8B Instruct	Extractive	58.4	(96.8)	57.7	(97.3)	60.6	(96.5)
SFT 8B	Extractive	59.8	(99.9)	57.7	(99.9)	61.5	(99.9)
CARL 8B (Ours)	Extractive	60.7	(100)	60.1	(99.9)	63.8	(99.9)
Llama 3.1 8B Instruct	General	59.0	(97.7)	59.0	(97.8)	61.1	(97.2)
SFT 8B	General	60.4	(99.2)	60.0	(99.2)	62.8	(99.1)
CARL 8B (Ours)	General	**61.6**	(98.9)	**61.3**	(99.0)	**64.4**	(98.8)

Table 17: Performance on datasets collected between 16 October 2024 and 30 November 2024, with F1-Recall (F1-R) and Answer Rate (AR, in %) as evaluation metrics.