🕐CHROKNOWLEDGE: UNVEILING CHRONOLOGICAL KNOWLEDGE OF LANGUAGE MODELS IN MULTIPLE DOMAINS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) have brought significant changes to many aspects of our lives. However, assessing and ensuring their chronological knowledge remains challenging. Existing approaches fall short in addressing the temporal adaptability of knowledge, often relying on a fixed time-point view. To overcome this, we introduce **CHROKNOWBENCH**, a benchmark dataset designed to evaluate chronologically accumulated knowledge across three key aspects: multiple domains, time dependency, temporal state. Our benchmark distinguishes between knowledge that evolves (e.g., personal history, scientific discoveries, amended laws) and knowledge that remain constant (e.g., mathematical truths, commonsense facts). Building on this benchmark, we present **CHROKNOWLEDGE** (Chronological Categorization of Knowledge), a novel sampling-based framework for evaluating LLMs' non-parametric chronological knowledge. Our evaluation led to the following observations: (1) The ability of eliciting temporal knowledge varies depending on the data format that model was trained on. (2) LLMs partially recall knowledge or show a cut-off at temporal boundaries rather than recalling all aspects of knowledge correctly. Thus, we apply our CHROKNOWPROMPT, an in-depth prompting to elicit chronological knowledge by traversing step-by-step through the surrounding time spans. We observe that it successfully recalls objects across both open-source and proprietary LLMs, demonstrating versatility, though it faces challenges with dynamic datasets and unstructured formats.

1 INTRODUCTION

Do large language models (LLMs) possess the ability to understand and track the history of knowledge as time progresses? In other words, can these models, which represent the cutting edge of modern artificial intelligence, reason appropriately about questions that involve evolving facts? Although some details remain controversial, knowledge—like science—is built upon accumulation (Zeigler, 2012; Picho et al., 2016). From raw data to information and to knowledge, every bit is cumulative which contributes to progress across all domains. This accumulation forms the foundation for higher-level reasoning, which is akin to wisdom in navigating the complexities of our world (Rowley, 2007). Given that LLMs are trained on vast and diverse corpora and are now integral to numerous applications in our daily lives, they must remain accurate and up-to-date to ensure reliability. Early versions of ChatGPT (OpenAI, 2022), for instance, sometimes produced inaccurate or absurd responses like the infamous example of "*The happening of King Sejong (1397-1450) throwing MacBook (2016-)*"[1]. These errors still give us a lesson that we need more precise model recalling knowledge correctly.

When we examine the issue closely, it's not just a matter of hallucination but also about whether the alignment of knowledge, particularly regarding dates, is accurate. Ensuring that LLMs maintain current and contextually relevant knowledge over time is crucial and researchers have explored various ways to investigate and verify the knowledge within these models (Zhang et al., 2024b). Pioneering works investigate whether language model has an ability of knowledge base or not in diverse domains (Petroni et al., 2019; Sung et al., 2021). Many subsequent studies analyze how LLMs define knowledge (Dai et al., 2022; Mishra et al., 2024); exploit how LLMs represent their

[1] https://english.hani.co.kr/arti/english_edition/e_international/1095956

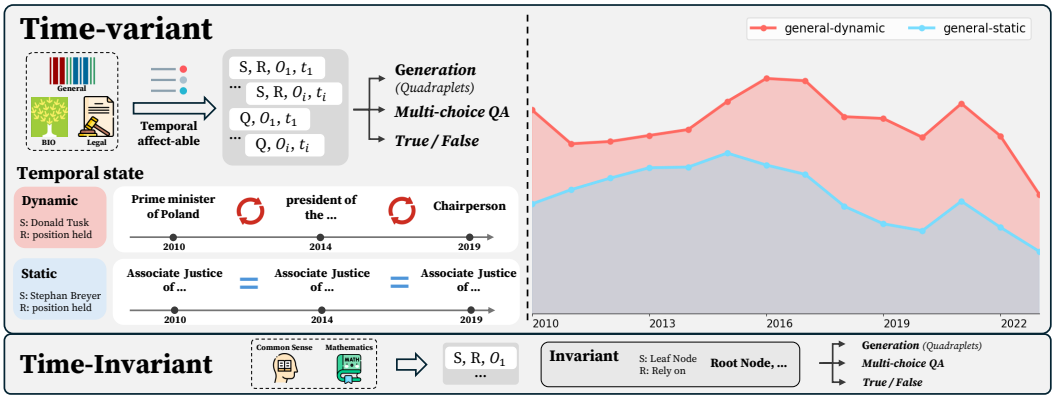

Figure 1: The overview of our **ChroKnowBench**. We gather knowledge with time stamps and separate them in three key aspects: (1) multiple domains: general, biomedical, legal, commonsense, and mathematics; (2) time dependency: as time goes by, changeable or not; (3) temporal state: dynamic (has evolved over period) and static (no change occurred during period).

knowledge (Geva et al., 2023; Zheng et al., 2024) with temporal context (Kasai et al., 2023; Fatemi et al., 2024); edit the misleading aspects of knowledge (Manakul et al., 2023; Wang et al., 2024c).

Here, we raise a question: "Do these methods sufficiently address the temporal adaptability of knowledge?". Current temporal-related approaches for evaluating and updating LLMs often focus on single time stamps, struggling to address the adaptable characteristics of knowledge over time (Jang et al., 2022a; Ge et al., 2024), which are especially important in specialized domains such as scientific discoveries and amended laws. This limitation can lead to outdated or incomplete information, undermining the models' effectiveness and safety. Addressing these challenges requires a comprehensive approach—temporal evolution, a core component of the knowledge accumulation.

Thus, we introduce **ChroKnowBench**, a benchmark dataset designed to evaluate chronologically accumulated knowledge across three dimensions: time dependency (time variant and time invariant), multiple domains (general, biomedical etc.), and temporal state (dynamic and static). ChroKnowBench differentiates between knowledge that is subject to evolution (e.g., transfer situation of a soccer player, scientific discoveries, and amended legal regulations) and knowledge that remains invariant (e.g., mathematical truths and commonsense facts). We then classify domains based on whether they are influenced by the flow of time, considering the domain specificity. Finally, we set the time frame to categorize the knowledge as either changeable or steady (Section 3).

Building on this benchmark, we also present **ChroKnowledge** (Chronological Categorization of Knowledge), a novel framework for assessing and enhancing the non-parametric chronological knowledge of LLMs. So, we start from analyzing how current open-source and proprietary LLMs work. As we expected, the time invariant knowledge shows steady in all time frame. However, for time variant dataset, the domain-specific characteristics significantly influence the representation of temporal knowledge from LLMs. More stable domains exhibit consistent performance, while more variable domains show more fluctuations. These observations highlight that we need a comprehensive approach to enhance representing temporal knowledge from LLMs (Section 5).

To this end, our ChroKnowPrompt approach utilizes an in-depth chronological prompting strategy that traverses knowledge across adjacent time spans, effectively addressing issues of partial recall and temporal boundaries (Section 6). In knowledge recall, our evaluation reveals improvements in the biomedical domain (11.2%) and the general domain (2.6%), shifting knowledge category from *Partial Known* to *Known* for unchanged objects. Our non-parametric approach allows for direct updates to both proprietary and open-source LLMs without extensive retraining, highlighting practicality and broad applicability (Section 7). Our work emphasizes the importance of temporal context in eliciting LLMs knowledge while identifying challenges with dynamic datasets and unstructured, context-rich formats like in the legal domain. A comprehensive analysis advocates for integrating parametric techniques to complement prompting and achieve more robust temporal knowledge handling.

Table 1: Knowledge categorization with a temporal component. We classify responses into *Correct*, *Partial Correct*, and *Incorrect* to specify eliciting predictions in diverse way by comparing them with the answer set A. We use a temperature set $\mathcal{T} \in 0, 0.7$ to capture variations in prediction, where $\mathcal{T}$ includes both greedy decoding and temperature sampling. We set n as 5, meaning that we evaluate using five distinct combinations of few-shot exemplars to ensure the robust assessment.

Category	Definition	Description
Correct	$\{\hat{o}_i \mid M(D_i, s, r, t) = \hat{o}_i; M, \tau = 0\}_{i=1}^{n} \subseteq A$	All objects generated with greedy decoding are entirely included within the answer set.
Partial Correct	$\bigcup_{\tau \in \mathcal{T}} \{\hat{o}_i \mid M(D_i, s, r, t) = \hat{o}_i; M, \tau\}_{i=1}^{n} \cap A \neq \emptyset$	At least one generated object from greedy decoding or temperature sampling is in the answer set.
Incorrect	$\bigcup_{\tau \in \mathcal{T}} \{\hat{o}_i \mid M(D_i, s, r, t) = \hat{o}_i; M, \tau\}_{i=1}^{n} \cap A = \emptyset$	None of the generated objects, either from greedy decoding or temperature sampling, are included in the answer set.

2 PRELIMINARIES

2.1 KNOWLEDGE CATEGORIZATION WITH A TEMPORAL COMPONENT

To distinguish and evaluate the knowledge levels of language models, we utilize the Sampling-based Knowledge Categorization (SliCK) framework (Gekhman et al., 2024). This approach starts by sampling the model M's answers to questions using various few-shot exemplar sets D. The sampling is conducted under two temperature conditions: $\tau = 0$ and $\tau > 0$. Then, it categorizes the degree to which the model knows each piece of knowledge into four levels: *HighlyKnown, MaybeKnown, WeaklyKnown*, and *Unknown*.

Based on Gekhman et al. (2024), we make the following modifications as follows: (1) We append a temporal component t to the conventional $\{subject\ (s), relation\ (r), object\ (o)\}$ triplet structure, allowing us to evaluate the model's knowledge across different time stamps; (2) We merge the two categories (*MaybeKnown* and *WeaklyKnown*) that represent recallable knowledge states[2] into a single category (*Partial Correct*); (3) By using time attribute, we also renamed the *HighlyKnown* and *Unknown* to the *Correct* and *Incorrect*, respectively. Our detailed definitions and descriptions are provided in Table 1. Although this setting allows us to categorize the model's sampled responses more precisely regarding time attribute, it only captures the model's knowledge at specific time points t, limiting our ability to observe changes over time. We address this limitation in Section 6.

2.2 ELICITING KNOWLEDGE USING DIVERSE TEMPLATES

Since models prefer different formats when eliciting their knowledge, it is important to use varied approaches to accurately assess their understanding (Zhou et al.). While we initially evaluate the model's knowledge using a standard triplet format, relying on a single template may not sufficiently capture the full extent of the model's knowledge. Thus, following Hendrycks et al. (2021); Huang et al. (2024), we also employ a well-known format, multiple-choice question answering (MCQA) with 4 options, and True/False to elicit the model's knowledge more effectively. As a result, we propose three templates for measuring how much knowledge the model holds: triplets (hereafter referred to as *Generation*), *MCQA*, and *TF*. Each template is designed with appropriate few-shot exemplars and corresponding matching rules. For example, in *Generation*, due to the complexity of evaluating responses, we apply fuzzy matching techniques to compare the generated responses against predefined labels. See Appendix A.2 for further details of few-shot exemplars, fuzzy matching rules, and examples of three templates.

3 CHROKNOWBENCH: CONSTRUCTING A BENCHMARK DATASET

In this section, we enumerate the details of constructing **CHROKNOWBENCH**, a chronologically accumulated knowledge benchmark dataset. The CHROKNOWBENCH dataset encompasses three key aspects: time dependency (time variant and invariant), multiple domains (general, biomedical, legal, commonsense, and math), and temporal state (dynamic and static). We first categorize knowledge into two groups: knowledge that remains unchanged over time (time invariant) and knowledge that

[2]We refer a recallable knowledge as the presence of at least one answer in the answer set, generated using either the greedy decoding or the temperature sampling method.

Table 2: Statistics of our benchmark dataset. We categorize whether knowledge changes over time (Time Variant) or remains constant (Time Invariant). Among five domains, we set the temporal state with dynamic (knowledge that changes within the time frame we have set) and static (knowledge that do not change within the time frame we have set). The number in parentheses represents the average change in objects per element within a dynamic dataset. See details in Appendix A.3.1.

Time Dependency	Domain (Time Frame)	# of Relations	Structured	Format	Temporal State	# of Examples	Source
Time Variant	general (2010 - 2023)	8	✓	(*s*, *r*, *o*, *t*)	dynamic (2.6)	8,330	Wikidata
					static	8,302	
	biomedical (2020 - 2024)	14	✓	(*s*, *r*, *o*, *t*)	dynamic (2.3)	7,345	UMLS
					static	7,345	
	legal (2010 - 2023)	6*	✗	QA	dynamic (1.1)	3,142	CFR
					static	3,142	
Time Invariant	commonsense	8	✓	(*s*, *r*, *o*)	invariant	24,788	CSKG
	math	12	✓	(*s*, *r*, *o*)	invariant	2,585	Math-KG

changes over time (time variant). Additionally, we classify domains based on whether they are influenced by the flow of time, considering the specificity of each domain. Finally, we categorize knowledge as either changeable (dynamic) or steady (static) within the time frame we have set.

3.1 Task Definition

Our primary focus is a time variant knowledge across three domains (general, biomedical, and legal) with comparisons to time invariant knowledge across two domains (commonsense and mathematics). What knowledge would be the difference between time variant and invariant? The time variant knowledge has a specific object changing across a stream of time. For example, "Cristiano Ronaldo *(s)* was a member of sports team of *(r)* Manchester United F.C. *(o_1)* in 2009 (t_1) and Real Madrid CF *(o_k)* in 2018 (t_k)". Likewise, we identify subject and object alias for each relation, then gather yearly changed objects. After accumulating object lists $\{o_1, o_2, \ldots, o_m\}$, we de-duplicate and fill out the missing data in specific years based on available data; objects between Manchester United F.C. *(o_1)* in 2009 (t_1) and Real Madrid CF *(o_k)* in 2018 (t_k), missing objects between 2010 (t_2) to 2017 (t_{k-1}) filled with existing object of 2009 (t_1). The statistic of ChroKnowBench dataset is in Table 2.

3.2 Dataset Generation

To construct dataset, we select annual knowledge sources for each domain, possible to be aligned with each elements even though the corpus does not specifically mention about time stamp. For sources with structured triplets, we identify temporal affect-able relations that typically change over time, such as *"position held"*. As time variant knowledge refers to the knowledge that has the potential to change over time, we divide it into two temporal states for more fine grained results: (1) **dynamic**, where the knowledge has evolved over the period. (2) **static**, where no change occurred during the period, though it has potential to be changed. Following the methodology outlined in Section 3.1, we track changes in objects to build the dynamic dataset, employing normalization and de-duplication for verification. Each object is checked with strict exact string match, then add into objects pool. Simultaneously, we identified unchanged objects over the same time frame to construct the static dataset. At the end, all data elements consist with an associated object pool $\{o_1, o_2, \ldots, o_m\}$ over time frames $\{t_1, t_2, \ldots, t_m\}$.

3.3 Time Variant & Invariant Knowledge

We sourced time variant knowledge from the general, biomedical, and legal domains. In general domain, we utilize Wikidata (Vrandečić & Krötzsch, 2014) dump to track object changes among the time frame using suggested time quantifiers. Collecting similar amounts of dynamic and static instances across eight relations, the result is formatted as $\{s, r, o, t\}$ quadruplet for each object and accompanied time stamp. For biomedical domain, we parse Unified Medical Language System (UMLS) (Bodenreider, 2004) metathesaurus data, where suggest yearly updated research in the domain, following previous work of BIOLAMA (Sung et al., 2021). Due to the slow pace of change in biomedical research, object pools in this domain shows slight expansions or contractions over time

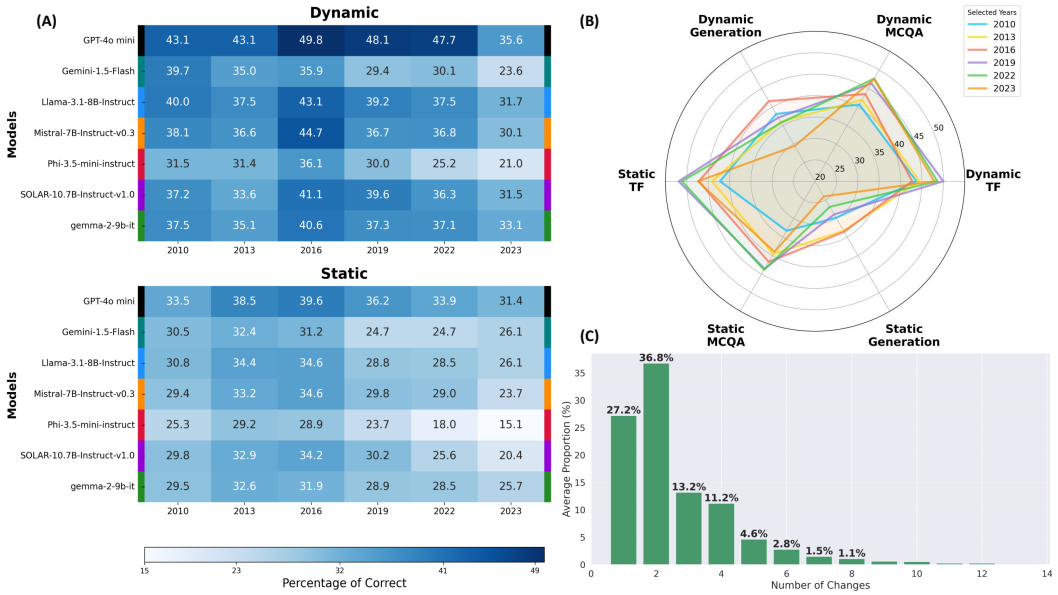

Figure 2: Performance analysis of general domain. (A) Heatmap in Generation template. For both dynamic and static datasets, a common trend across models is that performance is stronger in the intermediate years but decline recent years, reflecting the data-cutoff point. Dynamic knowledge shows more variation compared to static. Full results of total time frame is in Figure 10. (B) Template-wise performance for selected years. As time goes by, performance in generation goes low, on the other hand, MCQA and TF appeal to be rising. (C) Distribution of object changes in dynamic dataset.

frame. In the legal domain, we employ the Code of Federal Regulations (CFR) (U.S. Government) to track regulatory changes, as they suggest collection and accumulation of change in regulations at the end of year. Starting from pre-processing unstructured xml data, we adopt a QA-like format with placeholder for object, tracked among time frame which ends to dynamic or static whether it change or not. Time invariant knowledge, which remains constant regardless of temporal context, is drawn from common-sense and mathematical domains. We process the CSKG (Ilievski et al., 2021) dataset of commonsense knowledge, and Math-KG (Wang, 2022) for covering areas like data structures and pure mathematics. Further details are provided in Appendix A.3, especially the sources and the mechanism of object-level focus. Especially, compared to TKGs (Zhang et al., 2024a), which focus on temporal snapshots, our CHROKNOWBENCH emphasizes the detailed temporal evolution of individual knowledge elements, offering better adaptability for gradual changes; see Appendix A.3.3

4 EXPERIMENTAL SETUP

We conduct the experiments on a variety of proprietary and open-source LLMs. Each model utilizes either an instruction-tuned or chat version to enhance instruction following during sampling. We enumerate the eight open-source and two proprietary LLMs: Llama-3.1-8B-Instruct (Meta, 2024), Llama-3-8B-Instruct (Dubey et al., 2024), Llama-2-7b-chat-hf (Touvron et al., 2023b), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), Phi-3.5-mini-instruct (Abdin et al., 2024), SOLAR-10.7B-Instruct-v1.0 (Kim et al., 2023), gemma-2-9b-it (Team et al., 2024b), gemma-7b-it (Team et al., 2024a) for major open-source models, and GPT-4o mini (OpenAI, 2024a), Gemini-1.5-flash (DeepMind, 2024) for proprietary models. We focus on anlayzing trends in the chronological knowledge captured by those models, differ in corpus coverage. Details of our inference setups are in Appendix A.4.

5 CHROKNOWLEDGE: CHRONOLOGICAL CATEGORIZATION OF KNOWLEDGE

In this section, we introduce **CHROKNOWLEDGE** (***Chro**nological Categorization of **Knowledge***), a sampling-based framework designed to evaluate LLMs' non-parametric chronological knowledge. Our methodology assesses the temporal capabilities of LLMs using two distinct templates, detailed in Section 2.2, and explores how current LLMs encapsulate temporal information.

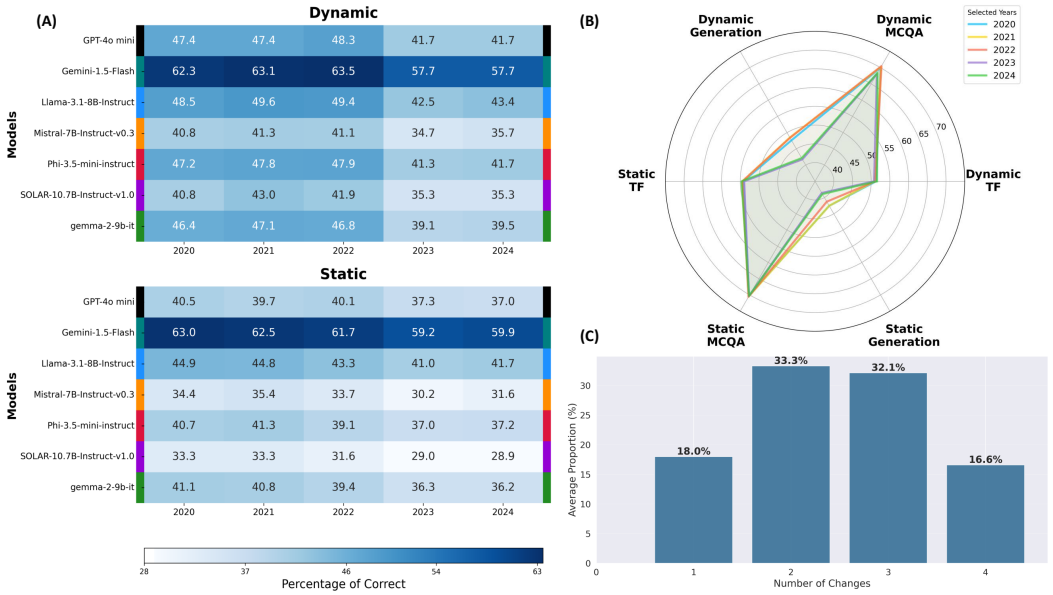

Figure 3: Performance analysis of biomedical domain. The format of figure is same as Figure 2. (A) Compared to the general domain, both dynamic and static datasets show lower variability, reflecting a domain-specific tendency toward consistency in knowledge changes. Both of them shows performance decrease between 2022 and 2023, aligning with the cutoff pattern noted in the general domain. (B) As time goes by, performance in generation declines, but MCQA and TF continue to perform well.

5.1 Results of Representing Knowledge from Large Language Models

For testing model's knowledge within the categorization, we sample five times for each knowledge to elicit it as possible in dynamic and static dataset. We present our findings across three different aspects: temporal-wise, template-wise, and domain-wise results. In Figure 2 and 3, we depict the results for all domains and especially address general and biomedical domains in main section. Legal and invariant datasets are presented in Appendix (Figure 4 and 9).

Temporal-wise Results. Comparing the upper and lower panels of (A) in Figure 2, 3 and 4 provides the tendency of temporal-wise results based on the generation results. A common trend across models is a decline in performance on recent knowledge, reflecting the pretraining corpus cutoff dates. Particularly in dynamic datasets, models demonstrate strong performance on earlier knowledge but experience a steeper decline in later periods, particularly in both general and biomedical domains. In contrast, static datasets show less fluctuation, with more stable yet weaker performance, highlighting limited temporal sensitivity due to reliance on a single timestamp. These results emphasize the need for frequent model updates, especially for dynamic knowledge, to ensure temporal relevance.

Template-wise Results. (B) of Figure 2, 3 and 4 provide the average scores of template-wise results, for more template specificity checking in three templates: generation, MCQA, and TF. Generation templates reveal a greater decline in recent knowledge, as models rely on internal information without predefined answers. In contrast, MCQA and TF templates help models select correct answers from structured options and predefined formats, mitigating some gaps in recent knowledge. This trend is more evident in the biomedical domain with MCQA templates and the legal domain with TF, revealing how domain-specific knowledge is more effectively elicited through specific formats. Also, dynamic dataset in the general domain is more sensitive to temporal shifts than static, highlighting the importance of task-specific templates in eliciting and improving temporal robustness.

Domain-wise Results. Comparing the Figure 2, 3 and 4 provide the tendency of domain-wise results, demonstrating distinct domain-specific characteristics of temporal knowledge change. In general domain, the models show a decline in recent knowledge, with a more unstable distribution of scores, which stems from domain's nature; changes in relations *'position held'* or *'member of sports team'* are more sensitive to temporal cues, leading to higher variability. Here, MCQA setting offers some resilience as it mitigates the knowledge decline observed in generation templates across

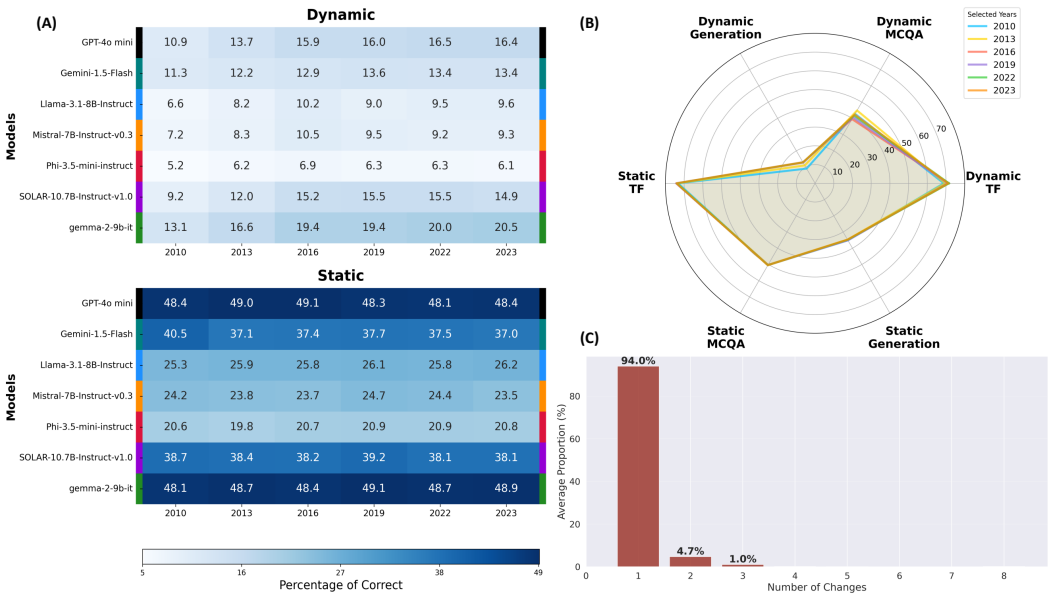

Figure 4: Performance analysis of legal domain. The format of figure is same as Figure 2. (A) Among time variant domains, legal domain shows the most stable results of static, while the gap between dynamic and static datasets is the largest among domains. (B) When it comes to each template, generation shows the lowest performance, while TF settings perform extraordinarily well in answering correctly. (C) In the legal domain, the distribution shows the lowest number of object changes over time, supporting the conclusion of the stable results in the heatmap.

dynamic and static datasets over different years. Every domain shows similar distribution of object changes with benchmark statistics, comparing (C) in each figure with Figure 8. This indicates that the models perform robustly as intended by the benchmark, without bias toward specific low changed objects. We provide more details for the overall trends of legal domain and time-invariant knowledge in Appendix A.5.

Overall, domain-specific characteristics significantly influence LLMs' temporal knowledge representation. More stable domains like biomedical and legal exhibit consistent performance with time invariant knowledge, while general domain shows more inconsistency. These insights underscore the need for tailored strategies to enhance LLMs' temporal knowledge capabilities.

6 CHROKNOWPROMPT: CHRONOLOGICAL KNOWLEDGE PROMPTING

6.1 CHRONOLOGICAL CATEGORIZATION

In previous section, we demonstrate whether open-source and proprietary models possess specific knowledge at various time stamps. However, this does not sufficiently assess the models' understanding of knowledge within a chronological progression. As Zhao et al. (2024) suggest, knowledge influenced heavily by temporal factors such as general domain can still vary in more stable situation like static dataset. To address this issue, we first reclassify the models' responses using a refined categorization scheme, allowing for a more comprehensive evaluation of temporal knowledge across all years.

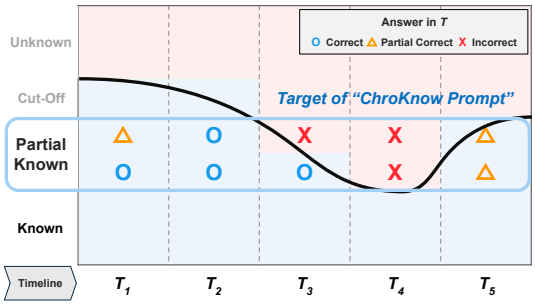

Figure 5: Chronological categorization based on each answer with its time stamp. If the model answer correctly for all, it is re-categorized as **Known**. The target of **ChroKnowPrompt** is **Partial Known**, which confuses its knowledge among the whole time stamps.

Figure 5 illustrates how it works: (1) **Known** for the precise temporal alignment if the model correctly identifies all relevant objects for a given

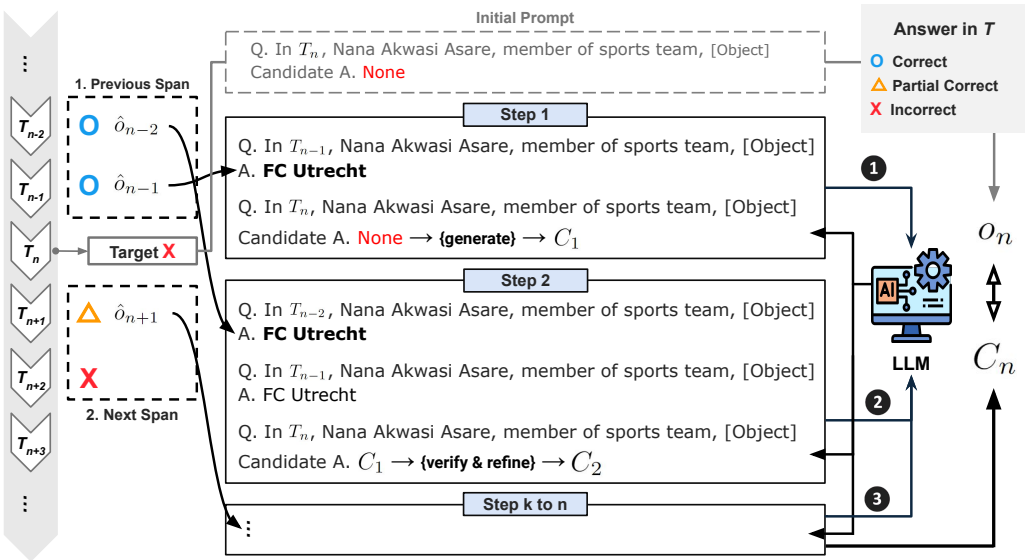

Figure 6: Overview of **ChroKnowPrompt**. The algorithm systematically traverses step by step, appending each span's result as few shot for each steps. The range of each previous and next span is predefined, with the order of nearest time stamp from target T_n. The model suggests last candidate answer C_n, verified and refined through several steps, which ends to be checked with the object o_n in benchmark.

knowledge category at each specific year; (2) If model fails to match the correct objects for every year, we refer it as **Unknown** for indicating incomplete or misaligned temporal knowledge; (3) The model accurately responds either just before or after a specific year but fails for others, signifying outdated information or forgotten legacy knowledge due to continuous updates (**Cut-off**); (4) The model correctly identifies some objects for a given year but not others, reflecting an incomplete understanding of the temporal knowledge (**Partial Known**).

Our main focus is on the **Partial Known** category, where models demonstrate substantial temporal knowledge but fail to answer correctly for all years, often showing confusion between correct answers. For example, Nana Akwasi Asare *(s)* was a member of sports team of *(r)* FC Utrecht *(o_n)* in 2011 *(t_n)*, but the model incorrectly identifies the team as FC Groningen, despite answering correctly with FC Utrecht for 2010 *(t_{n-1})* and in 2012 *(t_{n+1})*. At this point, we hypothesize that when the model gets one time stamp wrong, a more explicit focus on the temporal aspects surrounding that time span could help it generate more accurate answers. This is the core idea behind **ChroKnowPrompt**.

6.2 Method

We introduce a chronological prompting technique for non-parametric method to elicit chronological knowledge, aimed at bridging knowledge gaps by utilizing multiple temporal snapshots. This method enhances the model's reasoning by systematically integrating knowledge from different time stamps, enabling in-depth traverse. Our method is inspired by non-parametric editing techniques, such as Zhong et al. (2023) and Zheng et al. (2023), described in detail in Appendix A.1.

Figure 6 illustrates an example of the chronological prompting process. From a target year t_n, the algorithm systematically traverses the preceding years ($t_{n-1}, t_{n-2}, \ldots$) in the 'Previous Span' and the subsequent years ($t_{n+1}, t_{n+2}, \ldots$) in the 'Next Span'. For each traversed year, the most representative object $\hat{o}_k$ is selected from the *Correct* (represented by circle) and *Partial Correct* (represented by triangle) categories in Table 1 using majority voting.

Starting with the initial prompt containing the target time t_n, subject s_n, and relation r_n, the nearest year in the previous span is appended above the initial prompt with the selected object, forming the first step. The model then generates a candidate answer C_1 for t_n using this augmented prompt. Next, our ChroKnowPrompt iteratively adds prompts from progressively earlier years, refining the candidate answer by verifying its consistency across contexts (from C_1 to C_2). This backward traversal continues until a predefined span range is reached. Once the previous span is completed, our algorithm performs forward traversal by appending objects from subsequent years below the

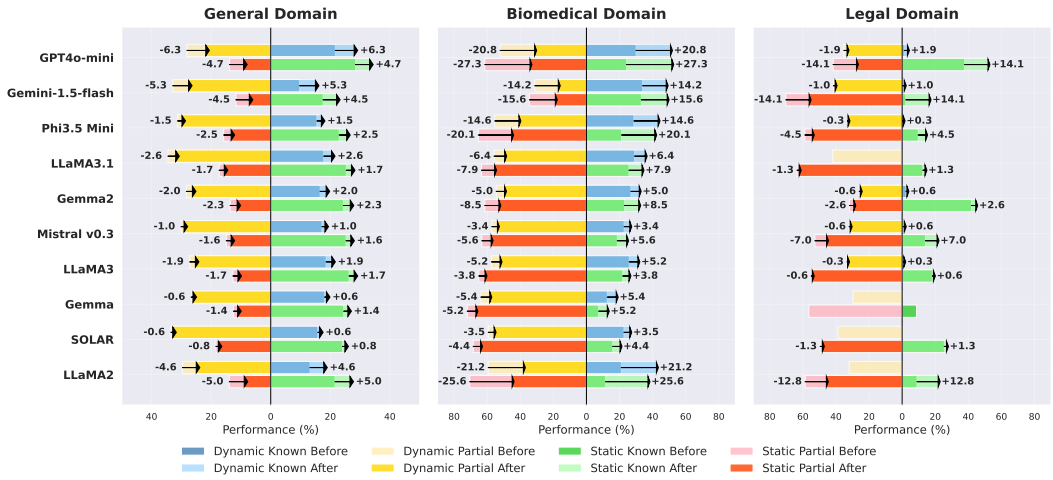

Figure 7: Results of **ChroKnowPrompt** across multiple domains with unchanged objects. For each domain, the left space represents the percentage of **Partial Known**, and the right represents the percentage of **Known**. Each model includes results for both dynamic (yellow-blue bar) and static (red-**green**) datasets, with arrows indicating the actual increase. As shown in plots, the most effective results are observed in the biomedical domain, where the unchangeable characteristic is stronger than the general domain. While the static dataset of the legal domain shows improvement, many models struggle with unstructured format, resulting in the lowest performance among the dynamic dataset.

target year, further generating and verifying candidate answers. If there are no previous or next years available, the process proceeds on only one side.

Upon completing all traversals, the final candidate answer C_n is compared against the benchmark object for t_n. If the candidate answer aligns correctly with the object for t_n, appropriately reflecting the temporal contexts, the knowledge categorization for t_n is updated to *Chrono-Correct*, which is equivalent to *Correct* for chronological assessments. In Appendix A.7, we provide the detailed steps.

7 Experimental Results & Analysis

7.1 Results of ChroKnowPrompt

Details of task configuration is in Appendix A.8. Figure 7 presents the effect of chronological prompting on unchanged objects across different models. Results are shown as the rise of percentage in *Known* category with decreasing *Partial Known*, indicating the increase by *Chrono-correct*. Significant overall improvements are observed in the biomedical domain (average increase of 11.2%), while general and legal domains show smaller gains (2.6% and 3.2%, respectively). This indicates that chronological prompting effectively enhances knowledge recall without requiring external retrieval systems. In the legal domain, models struggle to recall knowledge due to the complexity of unstructured, context-rich data, especially for dynamic dataset. Among the models, proprietary GPT4o-mini and Gemini-1.5-flash perform well, while open-source models like Llama-2-7b-chat-hf also show notable improvements despite outdated.

7.2 Analysis

Results in changed objects Table 5 compares results for changed and unchanged objects in dynamic datasets across domains. As shown in Section 7.1, **ChroKnowPrompt** effectively aids models in recalling unchanged objects in both dynamic and static datasets. However, its performance on changed objects in dynamic datasets remains limited, achieving only 10–30% performance of unchanged object cases (an average of 0.43 in general domains). This highlights, despite the helpfulness and applicability of **ChroKnowPrompt** in addressing the chronological gaps of knowledge, recalling all historical changes of objects using prompting alone remains an exceptionally challenging problem, particularly in complex contexts such as the legal domain. These findings emphasize the need for further research into effective parametric editing techniques to assist temporal knowledge handling.

Effects of chronological span To elucidate the mechanisms of chronological prompting, we analyze the impact of incorporating the next span in chronological contexts. As shown in Table 6, the

total span (both previous and next) yields higher scores than using only the previous span with the degree of improvement varying by domain. In the biomedical domain, the total span nearly doubles the score of the previous span alone (11.9 vs. 6.1), while the general domain shows a modest increase (2.8 vs. 1.8). Model-specific temporal sensitivity also varies: Llama-2-7b-chat-hf effectively utilizes next spans, whereas Gemini-1.5-flash and SOLAR-10.7B-Instruct-v1.0 benefit more from previous spans. These suggest that differences in temporal context utilization and the coverage of pretraining corpus may influence models' sensitivity and knowledge recall across time frames.

Effects of chat prompting Additionally, we analyze various chat models (before instruction-tuning), including Llama-2-7b-chat-hf with findings that chronological prompting can enhance both current and legacy models by leveraging temporal context. We evaluate three open-source chat models: mpt-7b-chat (Research, 2023), Pythia-Chat-Base-7B (Biderman et al., 2023), and nemotron-3-8b-chat-4k-sft-hf (Zhang et al., 2023a). As shown in Table 6 and 7, **ChroKnowPrompt** is not particularly effective for chat models. Only mpt-7b-chat achieves a comparable peformance to Llama-2-7b-chat-hf (an average increase of 11.4), while Pythia-Chat-Base-7B shows almost no improvement.

8 Related work

Since the emergence of LMs, deriving knowledge from language model is extensively studied, such as probing tasks (Hewitt & Manning, 2019), LAMA (Petroni et al., 2019) and BioLAMA (Sung et al., 2021). Then, many subsequent studies follows to exploit, (1) how LLMs define knowledge (Yu et al.; Zhang et al., 2023b; Gottesman & Geva, 2024), (2) how these models represent it (Chen et al., 2024a;b; Wang et al., 2024d), and (3) how manipulate misleading part (Wang et al., 2023; Gutiérrez et al., 2024; Wu et al., 2024a). Based on them, recent investigations of knowledge highlight the dynamic nature of evolving facts and suggests that contradictions within the training data may lead to knowledge conflicts (Marjanović et al., 2024; Chang et al., 2024; Wang et al., 2024a; Xu et al., 2024; Jin et al., 2024). And knowledge overshadowing (Zhang et al., 2024c) reveals phenomena where certain conditions overshadow other facts, leading to misleading information (i.e., hallucinations).

In other view point, exploring temporal knowledge starts from using Wikidata, a static format of knowledge in triplet: subject, relation, and object, originated from extracting literature-based knowledge (Hahn-Powell et al., 2017). Following pioneers like TimeQA (Chen et al., 2021) and TemporalWiki (Jang et al., 2022a), many works dealing with temporal and continuous knowledge flow (Zhang & Choi, 2021; Dhingra et al., 2022; Jang et al., 2022b; Liska et al., 2022; Nylund et al., 2023; Zhu et al., 2023; Khodja et al., 2024; Zhang et al., 2024d) consist in line of it. Building upon their achievement, CarpeDiem (Kim et al., 2024) emerges to simply identify whether knowledge is outdated or not, and DyKnow (Mousavi et al., 2024) maps various models' knowledge distribution. Also, (Zhao et al., 2024) makes dramatic work: align model into one fixed age. Though those impressive works, we try to broad the coverage of temporal knowledge. We utilize various templates to elicit the knowledge of LLMs, broaden the coverage of time stamps, and differentiate domains that should change based on a temporal perspective with those that should remain constant.

9 Conclusion, Limitation, and Future Work

Overall, our work highlights the critical role of temporal context in knowledge evaluation and introduces a framework for improving the temporal capabilities of future language models. We present **ChroKnowBench**, a benchmark for assessing temporal knowledge across diverse domains, and our **ChroKnowledge** framework, which evaluates LLMs' chronological knowledge through three types of templates. Our findings indicate that while models often recall facts with time stamps, they struggle with capturing full temporal boundaries, with reliance on rigid formats like MCQA and TF. By using ChroKnowPrompt, we improve knowledge recall by reducing ambiguous *Partial Known* and increasing *Known*, particularly in biomedical domains with strong performance on unchanged objects in both proprietary and open-source models. However, challenges persist in dynamic datasets and in unstructured, context-rich formats, which amplify the difficulty of capturing temporal evolution solely with prompting. Future work will focus on the need for parametric techniques to complement prompting, enabling better alignment with changing objects and complex temporal dependencies to enhance LLMs' temporal accuracy across various domains.

REFERENCES

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pp. 2397–2430. PMLR, 2023.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 2004.

Hoyeon Chang, Jinho Park, Seonghyeon Ye, Sohee Yang, Youngkyung Seo, Du-Seong Chang, and Minjoon Seo. How do large language models acquire factual knowledge during pretraining? *arXiv preprint arXiv:2406.11813*, 2024.

Jianhao Chen, Haoyuan Ouyang, Junyang Ren, Wentao Ding, Wei Hu, and Yuzhong Qu. Timeline-based sentence decomposition with in context learning for temporal fact extraction. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand, August 2024a. Association for Computational Linguistics.

Wenhu Chen, Xinyi Wang, and William Yang Wang. A dataset for answering time-sensitive questions. *arXiv preprint arXiv:2108.06314*, 2021.

Yuheng Chen, Pengfei Cao, Yubo Chen, Yining Wang, Shengping Liu, Kang Liu, and Jun Zhao. Cracking factual knowledge: A comprehensive analysis of degenerate knowledge neurons in large language models. 2024b.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. 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*, 2022.

Google DeepMind. Gemini flash: Lightweight models, two variants, both optimized for speed and efficiency. 2024.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Dan Gillick, Jacob Eisenstein, and William Cohen. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10, 2022.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of time: A benchmark for evaluating llms on temporal reasoning. *arXiv preprint arXiv:2406.09170*, 2024.

Xiou Ge, Ali Mousavi, Edouard Grave, Armand Joulin, Kun Qian, Benjamin Han, Mostafa Arefiyan, and Yunyao Li. Time sensitive knowledge editing through efficient finetuning. *arXiv preprint arXiv:2406.04496*, 2024.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning llms on new knowledge encourage hallucinations? *arXiv preprint arXiv:2405.05904*, 2024.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on EMNLP*, 2023.

Gaurav Rohit Ghosal, Tatsunori Hashimoto, and Aditi Raghunathan. Understanding finetuning for factual knowledge extraction. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 21–27 Jul 2024.

Daniela Gottesman and Mor Geva. Estimating knowledge in large language models without generating a single token. *arXiv preprint arXiv:2406.12673*, 2024.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models. *arXiv preprint arXiv:2405.14831*, 2024.

Gus Hahn-Powell, Marco A Valenzuela-Escárcega, and Mihai Surdeanu. Swanson linking revisited: Accelerating literature-based discovery across domains using a conceptual influence graph. In *Proceedings of ACL 2017, System Demonstrations*, 2017.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Advances in Neural Information Processing Systems*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=d7KBjmI3GmQ`.

John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Xiusheng Huang, Jiaxiang Liu, Yequan Wang, and Kang Liu. Reasons and solutions for the decline in model performance after editing. *arXiv preprint arXiv:2410.23843*, 2024.

Filip Ilievski, Pedro Szekely, and Bin Zhang. Cskg: The commonsense knowledge graph. In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*. Springer, 2021.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. Temporalwiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022a.

Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Jungkyu Choi, and Minjoon Seo. Towards continual knowledge learning of language models. In *10th International Conference on Learning Representations, ICLR 2022*. International Conference on Learning Representations, 2022b.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Li Qiuxia, and Jun Zhao. Tug-of-war between knowledge: Exploring and resolving knowledge conflicts in retrieval-augmented language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024.

Jaehun Jung, Jinhong Jung, and U Kang. T-gap: Learning to walk across time for temporal knowledge graph completion. *arXiv preprint arXiv:2012.10595*, 2020.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Velocity Yu, Dragomir Radev, Noah A Smith, Yejin Choi, and Kentaro Inui. Realtime qa: what's the answer right now? In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.

Hichem Ammar Khodja, Frédéric Bechet, Quentin Brabant, Alexis Nasr, and Gwénolé Lecorvé. Wikifactdiff: A large, realistic, and temporally adaptable dataset for atomic factual knowledge update in causal language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024.

Dahyun Kim, Chanjun Park, Sanghoon Kim, Wonsung Lee, Wonho Song, Yunsu Kim, Hyeonwoo Kim, Yungi Kim, Hyeonju Lee, Jihoo Kim, et al. Solar 10.7 b: Scaling large language models with simple yet effective depth up-scaling. *arXiv preprint arXiv:2312.15166*, 2023.

Yujin Kim, Jaehong Yoon, Seonghyeon Ye, Sangmin Bae, Namgyu Ho, Sung Ju Hwang, and Se-Young Yun. Carpe diem: On the evaluation of world knowledge in lifelong language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2024.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

Zixuan Li, Xiaolong Jin, Wei Li, Saiping Guan, Jiafeng Guo, Huawei Shen, Yuanzhuo Wang, and Xueqi Cheng. Temporal knowledge graph reasoning based on evolutional representation learning. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pp. 408–417, 2021.

Adam Liska, Tomas Kocisky, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, D'Autume Cyprien De Masson, Tim Scholtes, Manzil Zaheer, Susannah Young, et al. Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models. In *International Conference on Machine Learning*. PMLR, 2022.

Potsawee Manakul, Adian Liusie, and Mark Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9004–9017, 2023.

Sara Vera Marjanović, Haeun Yu, Pepa Atanasova, Maria Maistro, Christina Lioma, and Isabelle Augenstein. From internal conflict to contextual adaptation of language models. *arXiv preprint arXiv:2407.17023*, 2024.

Nick Mecklenburg, Yiyou Lin, Xiaoxiao Li, Daniel Holstein, Leonardo Nunes, Sara Malvar, Bruno Silva, Ranveer Chandra, Vijay Aski, Pavan Kumar Reddy Yannam, et al. Injecting new knowledge into large language models via supervised fine-tuning. *arXiv preprint arXiv:2404.00213*, 2024.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35, 2022.

Meta. Introducing llama 3.1: Our most capable models to date. 2024.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. Fine-grained hallucination detection and editing for language models. *arXiv preprint arXiv:2401.06855*, 2024.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. Fast model editing at scale. In *International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. Memory-based model editing at scale. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 17–23 Jul 2022.

Seyed Mahed Mousavi, Simone Alghisi, and Giuseppe Riccardi. Is your llm outdated? benchmarking llms & alignment algorithms for time-sensitive knowledge. *arXiv preprint arXiv:2404.08700*, 2024.

Kai Nylund, Suchin Gururangan, and Noah A Smith. Time is encoded in the weights of finetuned language models. *arXiv preprint arXiv:2312.13401*, 2023.

Yasumasa Onoe, Michael Zhang, Shankar Padmanabhan, Greg Durrett, and Eunsol Choi. Can lms learn new entities from descriptions? challenges in propagating injected knowledge. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023.

OpenAI. Introducing chatgpt. 2022.

OpenAI. Gpt-4o mini, advancing cost-efficient intelligence. 2024a.

OpenAI. Openai o1 system card. 2024b.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

Katherine Picho, Lauren A Maggio, and Anthony R Artino. Science: the slow march of accumulating evidence. *Perspectives on medical education*, 2016.

Mosaic AI Research. Introducing mpt-7b: A new standard for open-source, commercially usable llms. 2023.

Jennifer Rowley. The wisdom hierarchy: representations of the dikw hierarchy. *Journal of information science*, 2007.

Mujeen Sung, Jinhyuk Lee, S Yi Sean, Minji Jeon, Sungdong Kim, and Jaewoo Kang. Can language models be biomedical knowledge bases? In *2021 Conference on EMNLP 2021*. Association for Computational Linguistics (ACL), 2021.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024a.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024b.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

U.S. Government. Electronic code of federal regulations. URL `https://www.ecfr.gov/`.

Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 2014.

Jianing Wang. Math-kg: Construction and applications of mathematical knowledge graph. *arXiv preprint arXiv:2205.03772*, 2022.

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, et al. Knowledge mechanisms in large language models: A survey and perspective. *arXiv preprint arXiv:2407.15017*, 2024a.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *arXiv preprint arXiv:2405.14768*, 2024b.

Peng Wang, Ningyu Zhang, Bozhong Tian, Zekun Xi, Yunzhi Yao, Ziwen Xu, Mengru Wang, Shengyu Mao, Xiaohan Wang, Siyuan Cheng, Kangwei Liu, Yuansheng Ni, Guozhou Zheng, and Huajun Chen. EasyEdit: An easy-to-use knowledge editing framework for large language models. In Yixin Cao, Yang Feng, and Deyi Xiong (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, August 2024c. Association for Computational Linguistics.

Xiaohan Wang, Shengyu Mao, Ningyu Zhang, Shumin Deng, Yunzhi Yao, Yue Shen, Lei Liang, Jinjie Gu, and Huajun Chen. Editing conceptual knowledge for large language models. *arXiv preprint arXiv:2403.06259*, 2024d.

Yike Wang, Shangbin Feng, Heng Wang, Weijia Shi, Vidhisha Balachandran, Tianxing He, and Yulia Tsvetkov. Resolving knowledge conflicts in large language models. *arXiv preprint arXiv:2310.00935*, 2023.

Kevin Wu, Eric Wu, and James Zou. How faithful are rag models? quantifying the tug-of-war between rag and llms' internal prior. *arXiv preprint arXiv:2404.10198*, 2024a.

Xiaobao Wu, Liangming Pan, William Yang Wang, and Anh Tuan Luu. Updating language models with unstructured facts: Towards practical knowledge editing. *arXiv preprint arXiv:2402.18909*, 2024b.

Rongwu Xu, Zehan Qi, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. Knowledge conflicts for llms: A survey. *arXiv preprint arXiv:2403.08319*, 2024.

Lang Yu, Qin Chen, Jie Zhou, and Liang He. Melo: Enhancing model editing with neuron-indexed dynamic lora. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

Qinan Yu, Jack Merullo, and Ellie Pavlick. Characterizing mechanisms for factual recall in language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

David Zeigler. Evolution and the cumulative nature of science. *Evolution: Education and Outreach*, 2012.

Jinchuan Zhang, Bei Hui, Chong Mu, Ming Sun, and Ling Tian. Historically relevant event structuring for temporal knowledge graph reasoning. *arXiv preprint arXiv:2405.10621*, 2024a.

Michael Zhang and Eunsol Choi. Situatedqa: Incorporating extra-linguistic contexts into qa. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7371–7387, 2021.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, et al. A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*, 2024b.

Vivienne Zhang, Shashank Verma, Neal Vaidya, Abhishek Sawarkar, and Amanda Saunders. Nvidia ai foundation models: Build custom enterprise chatbots and co-pilots with production-ready llms. 2023a.

Yuji Zhang, Sha Li, Jiateng Liu, Pengfei Yu, Yi R Fung, Jing Li, Manling Li, and Heng Ji. Knowledge overshadowing causes amalgamated hallucination in large language models. *arXiv preprint arXiv:2407.08039*, 2024c.

Zhihan Zhang, Yixin Cao, Chenchen Ye, Yunshan Ma, Lizi Liao, and Tat-Seng Chua. Analyzing temporal complex events with large language models? a benchmark towards temporal, long context understanding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand, August 2024d. Association for Computational Linguistics.

Zihan Zhang, Meng Fang, Ling Chen, Mohammad-Reza Namazi-Rad, and Jun Wang. How do large language models capture the ever-changing world knowledge? a review of recent advances. In *Proceedings of the 2023 Conference on EMNLP*, pp. 8289–8311, 2023b.

Bowen Zhao, Zander Brumbaugh, Yizhong Wang, Hannaneh Hajishirzi, and Noah Smith. Set the clock: Temporal alignment of pretrained language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics ACL 2024*, Bangkok, Thailand and virtual meeting, August 2024. Association for Computational Linguistics.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on EMNLP*, 2023.

Danna Zheng, Mirella Lapata, and Jeff Z Pan. Large language models as reliable knowledge bases? *arXiv preprint arXiv:2407.13578*, 2024.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore, December 2023. Association for Computational Linguistics.

Yuxuan Zhou, Xien Liu, Chen Ning, and Ji Wu. Multifaceteval: Multifaceted evaluation to probe llms in mastering medical knowledge. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*. International Joint Conferences on Artificial Intelligence Organization. doi: 10.24963/ijcai.2024/737. URL https://doi.org/10.24963/ijcai.2024/737.

Xinyu Zhu, Cheng Yang, Bei Chen, Siheng Li, Jian-Guang Lou, and Yujiu Yang. Question answering as programming for solving time-sensitive questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

A Appendix

A.1 Supplementary Studies in Knowledge Editing

A.1.1 Parametric Knowledge Update

Considering update of LLMs are in two types, parametric and non-parametric (Wang et al., 2024c), a classical way of parametric update is using fine-tuning (Ghosal et al., 2024; Mecklenburg et al., 2024; Ge et al., 2024). While the method extended to LoRA (Hu et al.), QLoRA (Dettmers et al.), Melo (Yu et al., 2024) to continual learning (GRACE (Hartvigsen et al., 2024) and WISE (Wang et al., 2024b)), by using the parameter accessibility of white-box LLMs like Llama series (Touvron et al., 2023a), MEND (Mitchell et al.), ROME (Meng et al., 2022), MEMIT (Meng et al.) emerges. Those local editable methods are effective, and still try to improve specificity and generalizability.

A.1.2 Non-parametric Knowledge Update

In contrast, for black-box LLMs, updates rely on non-parametric knowledge methods (Onoe et al., 2023), such as SERAC (Mitchell et al., 2022), MeLLo (Zhong et al., 2023), and IKE (Zheng et al.,

2023). They align with two key trends: (1) Mitigating catastrophic forgetting, where the model loses previous knowledge, by not directly updating parameters. (2) Exploiting abilities of prominent black-box LLMs like GPT-o1 (OpenAI, 2024b) and Gemini (Team et al., 2023), as we cannot access to parameters. Another concern in knowledge update is they often focus only structured format, pointed out that current methods struggle to update unstrctured data effectively (Wu et al., 2024b). In this paper, we focus on non-parametric knowledge updates accomodating a broad range of input formats (structured and unstructured) to represent knowledge across diverse domains, depending on the use of various white-box and black-box LLMs.

A.2 Details of Eliciting Knowledge: Few-shot Exemplars, Fuzzy Matching Rules, and Examples of Three Templates

A.2.1 Few-shot Exemplars

To obtain the few-shot exemplar pool D, we leverage additional data collected using the same process as in ChroKnowBench. Specifically, for each individual relation type, We gather four exemplars for the general domain, and eight for the biomedical and legal domains. We then generate the few-shot exemplar set D_i by sampling four exemplars from D, which serves as actual demonstrations within a prompt. This process is repeated for every timestamp to ensure comprehensive temporal coverage.

A.2.2 Fuzzy Matching

We utilize the `rapidfuzz` library to compare the model's responses with the predefined labels. As the model's answer may a little bit different with complicated objects in specialized domains, such as the difference in order of words or upper and lower cases, using fuzzy match enables more rapid but still reliable quality without facilitating external NLI mechanisms.

Specifically, we employ a `token_set_ratio` metric with a threshold value set to 70 to determine a match. `token_set_ratio` is a metric used for comparing the similarity of two strings in a flexible manner, extending the functionality of the `token_sort_ratio`. In the preprocessing stage, the strings undergo tokenization, removal of punctuation, and conversion to lowercase. The tokens are then sorted in alphanumeric order before the similarity ratio is computed. This makes it useful for comparing strings where the word order may differ but the content is similar.

The key distinction of `token_set_ratio` lies in its incorporation of set operations, where duplicate words are removed. After eliminating repeated tokens, the same preprocessing steps as in `token_sort_ratio` are applied. When performing the comparison, the method checks if all tokens from the shorter string are contained within the longer string, making the approach particularly suited for cases where one string is a subset of the other. This flexible matching often results in higher accuracy for comparing strings with similar content but different structures, as illustrated by the example where a score of 100 is achieved when all tokens from the second string are present in the first.

A.2.3 Examples of Three Templates

We provide three templates of *Generation*, *MCQA*, and *TF* in the end of the Appendix for the better readability. For example, in Table 8 and Table 9, our target year is 2020 (*t*) to generate answer candidate of position held (*r*) by Donald Tusk (*s*).

A.2.4 Iterative Distractor Generation

For the Commonsense dataset, the objects corresponding to a given subject and relation are often ambiguous. When constructing compelling distractors, there is a higher likelihood (about 20%) of creating options that are actually correct answers rather than intended incorrect ones, compared to other datasets. Therefore, we include an additional verification process after generating the distractors, as outlined in Algorithm 1. Specifically, we formulate multiple-choice questions using the problem and the generated distractors, then use GPT-4o to select all correct answers. If it identifies more than one correct answer, we refine the distractors based on a prompt to recreate incorrect options.

Algorithm 1: Iterative Distractor Generation Algorithm

Data: Subject s, Relation r, Set of correct objects $\mathcal{O}_{\text{correct}}$
Result: Refined multiple-choice question q

```
Initialize conversation history H ← ∅
Initialize number of selected options n ← 1
while n > 0 do
    D ← LLMResponse(s, r, H)    // Generate three incorrect options
    q ← ComposeQuestion(s, r, D)    // Compose question using the generated distractors
    Append q to H
    ⊣ ← LLMResponse(q)    // LLM generates a response by solving the question
    S ← LLMResponse(⊣)    // Extract selected options using LLM
    n ← |S|    // Number of options selected by LLM
    if n > 0 then
        p ← CreatePrompt(s, r)    // Create another prompt for regenerating distractors
        Append p to H    // Add regeneration prompt to the conversation history
        D_new ← LLMResponse(s, r, H)    // Generate new set of distractors
        D[S] ← D_new[1 : n]    // Replace selected options
return q
```

A.3 Details of Benchmark Dataset

A.3.1 Statistics of Object Changes in Dynamic Dataset

The statistics of object changes among dynamic dataset in three time variant domains are in Figure 8. The figure shows how many objects have been changed among the time frame of each elements, and its distribution is proposed with the percentage of that elements across total number of dataset. The average number of object changes is 2.6 and 2.3 for general and biomedical dynamic dataset, while most of the element in legal domain has only one object changes among time frame.

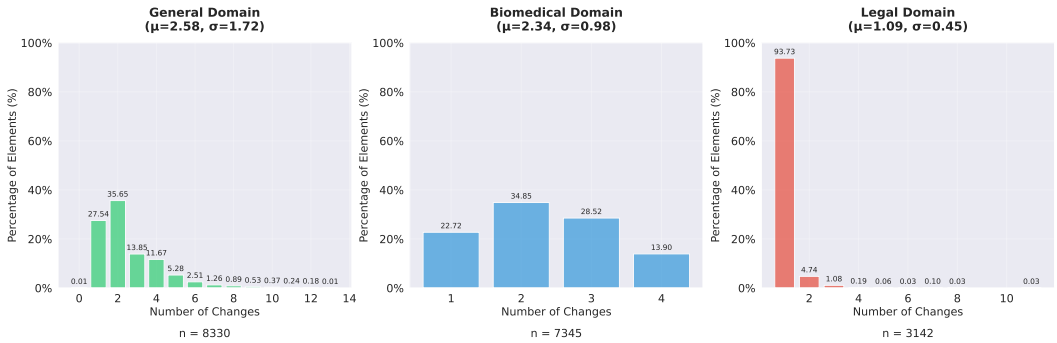

Figure 8: Statistics of object changes among dynamic dataset in three time variant domains.

A.3.2 Object-Level Focus in Benchmark Generation

Our ChroKnowBench emphasizes object-level changes as the core metric for assessing temporal knowledge dynamics. This approach reflects a deliberate balance between scalability, precision, and interpretability, aligning with the methodological goals of the benchmark.

Design and Scope The benchmark evaluates how well models can align and reason about temporal knowledge by focusing on object-level transformations at specific time points. For instance: Query 1: *"In 2001, Zidane was a player at Real Madrid."* and Query 2: *"In 2010, Zidane was a coach."* By treating these as distinct evaluations, ChroKnowBench isolates object-level transitions—such as roles or affiliations—while keeping the subject and relation fixed. This design ensures a structured and scalable evaluation process that captures significant, interpretable changes in temporal knowledge.

Reason of focusing on Object-Level Changes

- **Scalability:** The object-level focus simplifies the complexity of tracking temporal dynamics in subject-relation-object triples. By avoiding the combinatorial challenges associated with relational changes, this approach ensures clear and scalable evaluations.
- **Precision:** Object-level changes, such as shifts in roles or affiliations, capture more nuanced updates compared to relational changes, which often default to binary states (e.g., "is a player" → "is not a player"). This granularity enhances the depth of the evaluation.
- **Flexibility:** Unlike traditional Temporal Knowledge Graphs, which may impose rigid relational structures, the object-centered approach accommodates fine-grained changes in knowledge. This flexibility enables precise and interpretable assessments, particularly for dynamic, real-world transformations.

By centering evaluations on object-level changes, CHROKNOWBENCH delivers a robust framework for measuring temporal knowledge dynamics, balancing methodological rigor with practical scalability.

A.3.3 COMPARISON WITH TEMPORAL KNOWLEDGE GRAPHS

Temporal Knowledge Graphs (TKGs) are one of the well known approach for addressing temporal knowledge (Jung et al., 2020; Li et al., 2021; Zhang et al., 2024a). Our CHROKNOWBENCH also aim to model time-sensitive knowledge, but we adopt different approaches to structuring and interpreting temporal information. In this section, we compare these two paradigms across several dimensions, with particular attention to their handling of temporal snapshots, knowledge dynamics, and suitability for domains such as biomedical data.

Temporal Snapshot vs. Knowledge-Centric Tracking TKGs organize events based on temporal snapshots, incorporating multiple events occurring within the same timestamp into a single graph representation. This approach emphasizes the relationships between events at a specific time point, making it highly suitable for scenarios where the context of concurrent events is critical (e.g., *(North America, Host a visit, Business_Africa, 2010)*, *(Barack Obama, Consult, North America, 2010)*) (Zhang et al., 2024a). By connecting these events, TKGs enable reasoning about their inter-dependencies and broader temporal patterns.

In contrast, CHROKNOWBENCH adopts a knowledge-centric perspective, focusing on the temporal evolution of individual knowledge elements (s, r, o) over time. Instead of aggregating multiple events into a single snapshot, CHROKNOWBENCH tracks changes in object (o) values for each relation (r) across a timeline like the example in Section 3.1 This approach highlights the evolution of specific knowledge and ensures comprehensive tracking of its temporal progression.

Handling Specific Domains A key limitation of TKGs lies in their reliance on well-defined temporal snapshots. While this approach is effective for aggregating and reasoning about concurrent events, it becomes less suitable for domains like biomedical data, where changes often unfold gradually over time and are not tied to distinct temporal snapshots. For instance, the development of a medical treatment over a decade may involve incremental advancements that cannot be neatly encapsulated within discrete, event-based snapshots.

CHROKNOWBENCH overcomes this limitation not merely by dividing time into yearly intervals, but by prioritizing the temporal progression of individual knowledge elements. This distinction lies in its focus on tracking and organizing the dynamic and static changes of specific objects over time. While TKGs aggregate multiple concurrent events within a single snapshot, CHROKNOWBENCH constructs yearly object pools that capture the fine-grained evolution of a specific knowledge element across its temporal trajectory. These object pools allow CHROKNOWBENCH to explicitly track updates and fill gaps in data, ensuring a cohesive and complete representation of knowledge.

Furthermore, CHROKNOWBENCH incorporates the concept of dynamic and static datasets, categorizing knowledge based on its temporal variability. This approach enables detailed modeling of knowledge that evolves gradually while preserving distinctions from knowledge that remains unchanged over time. By avoiding the rigid aggregation of unrelated events and instead focusing on the chronological development of individual elements, CHROKNOWBENCH provides a more precise framework for fine-grained analysis in those specialized domains.

Table 3: Comparison of TKGs and CHROKNOWBENCH based on their temporal modeling approaches. TKGs focus on temporal snapshots aggregating multiple events, while CHROKNOWBENCH emphasizes tracking the temporal evolution of individual knowledge elements. This table highlights their strengths, weaknesses, and domain applicability.

Aspect	TKGs	CHROKNOWBENCH
Temporal Focus	Temporal snapshots aggregating multiple events	Temporal evolution of individual knowledge elements
Domain Applicability	Suitable for well-defined, event-rich domains (e.g., geopolitical, social networks)	Applicable not only temporal, but also gradual or implicit changes (e.g., biomedical, legal)
Handling of Gradual Changes	Limited by snapshot granularity	Effective through continuous tracking
Limitations	May overlook fine-grained changes in individual knowledge	May overlook broader event interdependencies

Comparative Summary TKGs excel at modeling inter-event relationships within temporal snapshots, making them effective for domains where concurrent event dependencies are critical, such as geopolitical analysis. However, they struggle with gradual changes and unstructured data, limiting their applicability in multiple domains. In contrast, CHROKNOWBENCH focuses on the detailed temporal evolution of individual knowledge units, leveraging object pools to capture gradual changes effectively, particularly in specialized domains like biomedical, and legal regulations.

Table 3 provides a comparative overview of the two paradigms.

A.3.4 SOURCE AND APPROACH OF BIOMEDICAL DOMAIN

In the biomedical domain, we follow previous work of BIOLAMA (Sung et al., 2021) framework to parse Unified Medical Language System (UMLS) yearly metathesaurus data. In the range of 2020 to 2024, we gather instances in 14 relations, resulting 7k for each dynamic and static dataset. Here, by considering domain specificity that the slow pace of change typical of long-term research, the object pool is slightly expanded or narrowed in that period; *Autonomic nerve structure*, with the relation *has indirect procedure site*, has a slightly broader scope in 2024, including additional objects like *Neurolytic autonomic nerve block* alongside previous objects such as *Intravenous regional autonomic block*. The format is same with general domain, $\{s, r, o, t\}$ quadruplet.

A.3.5 SOURCE AND APPROACH OF LEGAL DOMAIN

In the legal domain, we create a benchmark dataset based on the Code of Federal Regulations (CFR) from 2010 to 2023. We first extract paragraph-level data from regulatory documents for each year and employ Python's difflib library to detect changes between paragraphs across adjacent years (e.g., 2011 to 2012). Careful filtering is applied to ensure that only paragraphs with minor modifications (e.g., single-word updates or subtle phrasing changes) are retained.

To further analyze the dataset, we utilize the spaCy en_core_web_lg model to detect named entities in the paragraphs and assess whether these changes involve modifications to the detected entities. Despite noise introduced by the NER model, we initially identify around 56K changes for near-year comparisons. These changes are grouped into sequences of years to track alterations over time, while filtering out paragraphs that are introduced or removed in intermediate years. Ultimately, we focus on paragraphs present in all years between 2010 and 2023, resulting in 8,793 paragraphs.

We then apply GPT-4o-mini to assess whether the detected changes are semantically meaningful, excluding minor corrections like typographical fixes or abbreviations. This results in a refined set of 4,362 meaningful updates. Additionally, we select 4,746 unchanged paragraphs containing entities detected by the NER model. For each paragraph, we format the changes as fill-in-the-blank tasks, where the modified part is replaced with a blank, providing a rich resource for studying legal text evolution over time.

A.3.6 SOURCE AND APPROACH OF COMMONSENSE AND MATHEMATICS

In the commonsense domain, we utilized the CSKG dataset presented in the CSKG paper. Unlike the BIO dataset, the object lists for each triplet in this dataset consist of synonymous terms, allowing multiple triplets to share the same subject and relation. In such cases, the objects appearing in each triplet carry distinct meanings. Out of the 6 million triplets, we merged the objects of triplets that have the same subject and relation into a single set, and then sampled x number of triplets from this collection.

In the mathematics and data structure/algorithm domain, we utilized the Math-KG dataset introduced in the Math-KG paper. This dataset, originally in Chinese, stores multiple objects with the same subject and relation across different triplets. Each object was translated into English using GPT-4, after which the objects from triplets sharing the same subject and relation were merged to construct a final dataset consisting of 22k triplets.

For the CommonSense dataset, the answers (objects) corresponding to a given subject and relation are often ambiguous. Consequently, when constructing compelling distractors, there is a higher likelihood (about 20%) of creating options that are actually correct answers rather than intended incorrect ones, compared to other datasets. Therefore, we include an additional verification process after generating the distractors, as outlined in Algorithm 2. Specifically, we formulate multiple-choice questions using the problem and the generated distractors, then ask GPT-4o to select all correct answers. If it identifies more than one correct answer, we refine the distractors based on a prompt to recreate incorrect options.

A.4 INFERENCE SETTING

We evaluates all models using vLLM (Kwon et al., 2023) system, supporting features of efficient KV cache memory management that dramatically decrease inference time. All white box LM inference is conducted by vLLM with hyper-parameter: BFloat16, fixed seed, two kinds of temperature based on each sampling setting(greedy decoding with 0.0, and high temperature with 0.7). The precision is done with eight NVIDIA A100 GPUs(80GB).

A.5 DETAILS OF **CHROKNOWLEDGE** IN LEGAL AND TIME-INVARIANT DOMAIN

Figure 4 shows the result of legal domain. Among time variant domains, legal domain shows the most stable results of static, also minimal decline in dynamic dataset. This indicates the domain specificity, which has less frequent yearly changes (almost cases has one change of object in total time frame) and change continues across many time stamps. Also, model's capability for specific task setting is influential in legal domain like the result of MCQA and TF shows, where gap between generation and other templates are many times larger than in other domains.

For commonsense and mathematics in Figure 9, arbitrary years based on the biomedical domain were used, from 2020 to 2024. The left side of result shows the tendency of generation templates, and the middle side is the tendency of MCQA templates, and the last one for TF templates. Each results measure the percentage of *Correct* answer, represented as line plots. Results show minimal variation, aligning with the stable nature of these knowledge types. This consistency confirms that time-invariant knowledge is well-preserved across models. For template wise comparison, generation cases show a way little gap between models, while MCQA tasks show the different between models, which is aligned with the findings from other time variant domains: the ability of each model's specialized task affects its knowledge recall ability.

About the overall performance quality, the result of time invariant shows lower performance as the models generate one object per each knowledge, while the time invariant knowledge's coverage is wider than other domains. This tendency is alleviated by using MCQA and TF templates, which ends of the rationale for helpfulness of using multiple templates to check knowledge.

A.6 TOTAL TIME FRAME RESULT OF **CHROKNOWLEDGE**

Figure 10 represents the total results for all the range of time frames in general, and legal domain. Figure 11 to Figure 15 represents the total results of template-wise performance in **ChroKnowledge**. Each domain's result is separated into two category of temporal state: Dynamic and Static. Every

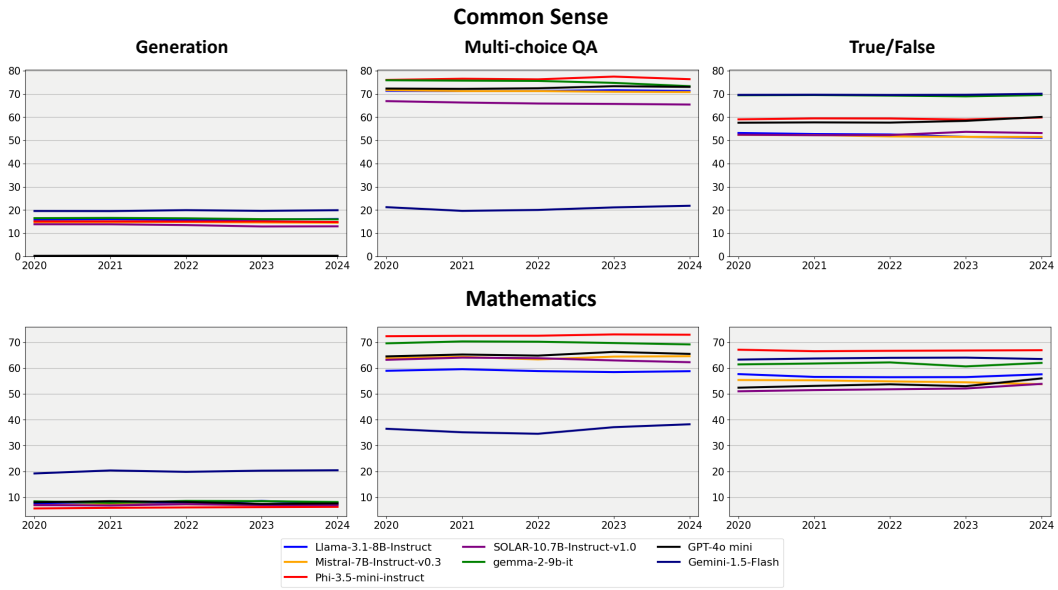

Figure 9: Performance analysis of common-sense and mathematics domains. Three line plots represent each template's results: Generation, MCQA and TF. All model shows clearly the domain specific characteristics, which is invariant knowledge even it comes with temporal attributes. Overall results are lower in generation templates, as it is challenging for models to correctly recall exactly one object in these domains (e.g., 'subject': 'Parent', 'relation': 'Synonym' has more objects later than 'Ancestor')

spider plots consist with three template: Generation, Multi-choice QA, and True/False. Each statistics refer the percentage of *Correct* answer, same as Figure 2.

A.7 Algorithm of **ChroKnowPrompt**

The overall scheme of ChroKnowPrompt is down below. As described in Section 6.2, the algorithm starts from making initial prompt with target time t_n, subject s_n and relation r_n from target triplet. As the initialized candidate answer is None that model cannot properly answer for that target year, the algorithm also starts with make a empty list of candidate answer list A. and accumulated prompt $\mathcal{P}$. Then, the algorithm checks the correct object within each time span P and N. If one of those span has no correct object, the algorithm passes that side of traversal. It the preparation is all done, the first step in previous span (if no previous span exists, the nearest next span) begins with selecting object $\hat{o}$ by majority voting. Appending prompts in each step, the model is asked to generate or verify and refine the answer C of each step, like in Figure 6. After all step is done, the last candidate answer, which is the most refined result, is being checked with the original target object o_n coming from target triplet. If it is matched (we also used fuzzy match in here), the category of *Incorrect* is updated to *Chrono-correct*.

Algorithm 2: Chronological Prompting Algorithm

Data: Correct set $\mathcal{C} = \{(t_i, c_i)\}$, target time t, triplet (s, r, o), Prev span P, Next span N
Result: List of candidate answers $\mathcal{A}$, Updated Category

Initialize accumulated prompt $\mathcal{P} \leftarrow \emptyset$
Initialize candidate answer $a \leftarrow \emptyset$
Initialize candidate answer list $\mathcal{A} \leftarrow \emptyset$
$\mathcal{T}_{\text{prev}} \leftarrow$ time before t in $\mathcal{C}$ up to span P // Find correct object in previous time
$\mathcal{T}_{\text{next}} \leftarrow$ time after t in $\mathcal{C}$ up to span N // Find correct object in next time
if $\mathcal{T}_{prev} = \emptyset$ **then**
  Skip backward traversal and process next years only
if $\mathcal{T}_{next} = \emptyset$ **then**
  Skip forward traversal and process previous years only
for $t_p \in \mathcal{T}_{prev}$ **do** // Process previous years first
  $\hat{o} \leftarrow \text{MajorityVote}(\mathcal{C}(t_p))$ // Get the correct object by majority voting
  $\mathcal{M} \leftarrow \text{PromptAugment}(t_p, t, s, r, \mathcal{P}, \hat{o}, a, \text{'previous'})$ // Augment prompt by adding above
  $a_{\text{new}} \leftarrow \text{LLMResponse}(\mathcal{M})$ // Generate or verify answer based on system prompt
  $a_{\text{ext}} \leftarrow \text{ExtractAnswer}(a_{\text{new}})$
  if $a_{ext} \neq \emptyset$ ***and*** $a_{ext} \neq a$ **then**
    $a \leftarrow a_{\text{ext}}$
  Append a to $\mathcal{A}$
  Update accumulated prompt $\mathcal{P}$ with $\mathcal{M}$
for $t_n \in \mathcal{T}_{next}$ **do** // Process next years after previous years
  $\hat{o} \leftarrow \text{MajorityVote}(\mathcal{C}(t_p))$ // Get the correct object by majority voting
  $\mathcal{M} \leftarrow \text{PromptAugment}(t_n, t, s, r, \mathcal{P}, \hat{o}, a, \text{'next'})$ // Augment prompt by adding below
  $a_{\text{new}} \leftarrow \text{LLMResponse}(\mathcal{M})$ // Generate or verify answer based on system prompt
  $a_{\text{ext}} \leftarrow \text{ExtractAnswer}(a_{\text{new}})$
  if $a_{ext} \neq \emptyset$ ***and*** $a_{ext} \neq a$ **then**
    $a \leftarrow a_{\text{ext}}$
  Append a to $\mathcal{A}$
  Update accumulated prompt $\mathcal{P}$ with $\mathcal{M}$
if $\forall a_i \in \mathcal{A}, a_i = o$ **then**
  Update knowledge categorization to *Chrono-Correct*
return $\mathcal{A}$, *Updated Category*

A.8 TASK CONFIGURATIONS OF CHROKNOWPROMPT

We apply our method to both *Incorrect* and *Partial Correct* categories, as the latter may still lack definitive answers. The test set consists of 10% of the total dataset from each domain. Evaluation employs fuzzy matching with a temperature of 0 for strict assessment, classifying an answer as *Chrono-correct* only if the last candidate answer matches the object. As described in Section 6.2, the system prompt for each case (generation or verification&refinement) works as follows Table 4:

Generation Case
[System] Answer 'Candidate A. [Object]' based on the timestamp. Output only the answer: 'A. [Object]'.
Verify & Refine Case
[System] Answer 'Candidate A. [Object]' based on the timestamp. If it is correct, repeat the same [Object]. If it is wrong, generate a new [Object]. Output only the answer: 'A. [Object]'.

Table 4: System prompts for **ChroKnowPrompt**.

Table 5: Result of **ChroKnowPrompt** for both object changed and unchanged cases. The order of open-sources LLM is sorted by release date, starting from the latest model to the most outdated model. The numeric score represents the level of **Known increase** in chronological categorization, and the increase is due to the transition of previously confusing *Partial correct* responses to *Chrono-correct*. The parenthesis score is the total percentage in dynamic or static dataset, including both changed and unchanged cases. Showing almost 10% to 30% performance of object unchanged cases, the results gives observations that only prompting method has limitations in editing diversely changing temporal knowledge.

Models	general			biomedical			legal		
	dynamic		static	dynamic		static	dynamic		static
Object	changed	unchanged	unchanged	changed	unchanged	unchanged	changed	unchanged	unchanged
Proprietary Large Language Models									
GPT4o-mini	+0.7 (28.7)	+**7.0** (28.7)	+4.7 (33.2)	+1.1 (51.9)	+21.9 (51.9)	+**27.8** (51.6)	+0.0 (3.2)	+**1.9** (3.2)	+**14.1** (51.9)
Gemini-1.5-flash	+0.6 (15.6)	+5.9 (15.6)	+4.5 (22.1)	+0.8 (49.0)	+15.0 (49.0)	+16.0 (48.8)	+0.0 (1.3)	+1.0 (1.3)	+14.1 (16.3)
Open-Source Large Language Models									
Phi3.5 Mini	+0.3 (17.3)	+1.8 (17.3)	+2.5 (25.5)	+**2.1** (45.4)	+16.7 (45.4)	+20.3 (41.3)	+0.0 (0.6)	+0.3 (0.6)	+4.5 (14.2)
LLaMA3.1	+0.2 (20.6)	+2.8 (20.6)	+1.7 (27.1)	+1.4 (36.9)	+7.8 (36.9)	+7.9 (33.6)	+**0.0** (0.3)	+**0.0** (0.3)	+1.3 (13.8)
Gemma2	+**1.0** (19.6)	+3.0 (19.6)	+2.3 (26.7)	+0.6 (32.5)	+5.6 (32.5)	+9.0 (31.7)	+0.0 (2.9)	+0.6 (2.9)	+2.6 (44.6)
Mistral v0.3	+0.4 (18.6)	+1.5 (18.6)	+1.6 (26.9)	+0.4 (26.6)	+3.8 (26.6)	+5.6 (24.3)	+0.0 (1.3)	+0.6 (1.3)	+7.0 (21.1)
LLaMA3	+0.4 (20.9)	+2.3 (20.9)	+1.7 (28.0)	+0.3 (31.4)	+5.5 (31.4)	+**3.8** (25.7)	+0.0 (1.0)	+0.3 (1.0)	+0.6 (18.9)
Gemma	+0.2 (18.9)	+0.8 (18.9)	+1.5 (25.9)	+**0.3** (18.3)	+5.7 (18.3)	+5.3 (12.6)	+**0.0** (0.3)	+**0.0** (0.3)	+**0.0** (8.70)
SOLAR	+**0.1** (16.5)	+**0.7** (16.5)	+**0.9** (24.9)	+0.3 (26.5)	+**3.8** (26.5)	+4.5 (20.3)	+0.0 (0.6)	+0.0 (0.6)	+1.3 (26.8)
LLaMA2	+0.3 (18.1)	+4.9 (18.1)	+**5.0** (26.6)	+2.0 (44.3)	+**23.2** (44.3)	+26.3 (37.2)	+**0.0** (0.3)	+**0.0** (0.3)	+12.8 (21.8)
Object Increase	0.4	2.9		0.9	**11.8**		**0.0**	3.2	
Temporal Increase	1.7	2.6		5.9	**12.7**		**0.2**	5.8	
Domain Increase	**2.0**			**8.2**			2.1		

Table 6: Result of **ChroKnowPrompt** in span-wise comparison for general and biomedical domain. The order of open-sources LLM is sorted by release date, starting from the latest model to the most outdated model. The numeric score is the level of **Known** in chronological categorization and the increase in parentheses is from the ratio of *Chrono-correct* which was confusing *Partial correct* before. Each result presents both in total span and previous span.

Models	general				biomedical				Model Increase	
	total span		previous span		total span		previous span		total span	previous span
	dynamic	static	dynamic	static	dynamic	static	dynamic	static		
Proprietary Large Language Models										
GPT4o-mini	28.7 (**+7.7**)	33.2 (+4.7)	26.6 (+5.7)	31.7 (+3.3)	51.9 (+23.0)	51.6 (+27.8)	41.8 (+12.8)	36.7 (+13.0)	**15.8**	8.7
Gemini-1.5-flash	15.6 (+6.5)	22.1 (+4.5)	15.3 (**+6.1**)	21.7 (**+4.1**)	49.0 (+15.8)	48.8 (+16.0)	48.0 (**+14.9**)	51.7 (**+18.8**)	10.7	**11.0**
Open-Source Large Language Models										
Phi3.5 Mini	17.3 (+2.1)	25.5 (+2.5)	16.5 (+1.2)	24.1 (+1.1)	45.4 (+18.7)	41.3 (+20.3)	36.6 (+10.0)	31.5 (+10.5)	10.9	5.7
LLaMA3.1	20.6 (+3.1)	27.1 (+1.7)	19.4 (+1.9)	26.4 (+1.0)	36.9 (+9.2)	33.6 (+7.9)	32.0 (+4.2)	29.1 (+3.4)	5.5	2.6
Gemma2	19.6 (+4.0)	26.7 (+2.3)	17.8 (+2.2)	24.7 (+0.4)	32.5 (+6.2)	31.7 (+9.0)	27.9 (+1.5)	26.7 (+4.1)	5.4	2.1
Mistral v0.3	18.6 (+1.8)	26.9 (+1.6)	18.3 (+1.6)	26.8 (+1.5)	26.6 (+4.2)	24.3 (+5.6)	24.6 (+2.2)	21.3 (+2.6)	3.3	2.0
LLaMA3	20.9 (+2.7)	28.0 (+1.7)	20.8 (+2.5)	27.2 (+0.9)	31.4 (+5.7)	25.7 (+3.8)	28.7 (+3.0)	24.2 (+2.3)	3.5	2.2
Gemma	18.9 (+1.0)	25.9 (+1.5)	18.8 (+0.8)	25.3 (+0.8)	18.3 (+6.0)	12.6 (+5.3)	16.0 (+3.7)	9.60 (+2.3)	3.5	1.9
SOLAR	16.5 (+0.8)	24.9 (+0.9)	16.7 (+1.1)	25.1 (+1.1)	26.5 (+4.1)	20.3 (+4.5)	27.7 (+5.3)	19.7 (+3.8)	2.6	2.8
LLaMA2	18.1 (+5.2)	26.6 (**+5.0**)	15.9 (+3.0)	23.1 (+1.5)	44.3 (**+25.2**)	37.2 (+26.3)	32.5 (+13.4)	23.3 (+12.4)	15.4	7.6
Open-Source Chat Models										
Mpt	18.3 (+4.8)	25.6 (+4.8)	17.0 (+3.5)	22.8 (+2.1)	43.3 (+22.9)	45.3 (**+30.3**)	30.8 (+10.4)	26.6 (+11.6)	15.7	6.9
Pythia	13.8 (**+0.0**)	20.8 (**+0.1**)	13.8 (**+0.0**)	20.7 (**+0.0**)	13.1 (**+0.0**)	10.2 (**+0.1**)	13.1 (**+0.0**)	10.2 (**+0.1**)	**0.1**	**0.0**
Nemotron3	11.2 (+1.5)	18.3 (+1.8)	10.1 (+0.5)	16.7 (+0.1)	22.1 (+9.0)	19.4 (+8.3)	17.9 (+4.8)	15.4 (+4.4)	5.2	2.5
Temporal Increase	3.2	2.5	2.3	**1.4**	11.5	**12.7**	6.6	6.8		
Domain Increase	2.9		**1.8**		**12.1**		6.8			

A.9 DETAILS IN SPAN-WISE RESULTS OF CHROKNOWPROMPT

Table 6 and 7 present the evaluation of **ChroKnowPrompt** in span-wise comparisons. While our approach demonstrates significant improvements in certain domains, it shows limited or negligible gains in the legal domain. Overall scores in the dynamic dataset remain modest, with the highest gain being only 1.9. However, the static dataset yields more impressive results, with the highest increase exceeding 10% in proprietary models, a level comparable to the biomedical domain's results. Another finding is that although the increase in Table 6's result in general domain is not higher than the static figures in the legal domain, the variation in figures between models is significantly larger in the legal domain. As the format of legal dataset is the unstructured format with long context, this would be one factor of low edit quality.

Table 7: Result of **ChroKnowPrompt** in span-wise comparison for legal domain. The order of open-source LLMs follows the same sequence as in Table 6, starting with the latest model and progressing to the most outdated one. The numeric score represents the level of **Known** in chronological categorization, and the increase in parentheses reflects the ratio of *Chrono-correct* answers, considering total span in the left side and previous span in the right side.

Models	legal				Model Increase	
	total span		previous span		total span	previous span
	dynamic	static	dynamic	static		
Proprietary Large Language Models						
GPT4o-mini	3.2 **(+1.9)**	51.9 **(+14.1)**	2.6 **(+1.3)**	48.4 (+10.6)	**8.0**	6.0
Gemini-1.5-flash	1.3 (+1.0)	16.3 (+14.1)	1.6 (+1.3)	18.5 **(+16.3)**	7.6	**8.8**
Open-Source Large Language Models						
Phi3.5 Mini	0.6 (+0.3)	14.2 (+4.5)	0.6 (+0.3)	11.9 (+2.3)	2.4	1.3
LLaMA3.1	0.3 **(+0.0)**	13.8 (+1.3)	0.3 **(+0.0)**	12.5 (+0.0)	0.7	**0.0**
Gemma2	2.9 (+0.6)	44.6 (+2.6)	2.6 (+0.3)	43.9 (+1.9)	1.6	1.1
Mistral v0.3	1.3 (+0.6)	21.1 (+7.0)	1.0 (+0.3)	19.2 (+5.1)	3.8	2.7
LLaMA3	1.0 (+0.3)	18.9 (+0.6)	1.3 (+0.6)	18.9 (+0.6)	0.5	0.6
Gemma	0.3 **(+0.0)**	8.70 (+0.0)	0.3 **(+0.0)**	8.70 (+0.0)	**0.0**	**0.0**
SOLAR	0.6 (+0.0)	26.8 (+1.3)	0.6 (+0.0)	28.4 (+2.9)	0.7	1.5
LLaMA2	0.3 **(+0.0)**	21.8 (+12.8)	0.3 **(+0.0)**	17.3 (+8.3)	6.4	4.2
Open-Source Chat Models						
Mpt	1.0 (+0.6)	8.4 (+5.1)	0.6 (+0.3)	4.5 (+1.3)	2.9	0.8
Pythia	0.3 **(+0.0)**	3.2 **(+0.0)**	0.3 **(+0.0)**	3.2 **(+0.0)**	**0.0**	**0.0**
Nemotron3	0.3 **(+0.0)**	5.1 (+1.0)	0.3 **(+0.0)**	4.8 (+0.6)	0.5	0.3
Temporal Increase	0.4	**5.0**	**0.3**	3.8		
Domain Increase	**2.7**		**2.1**			

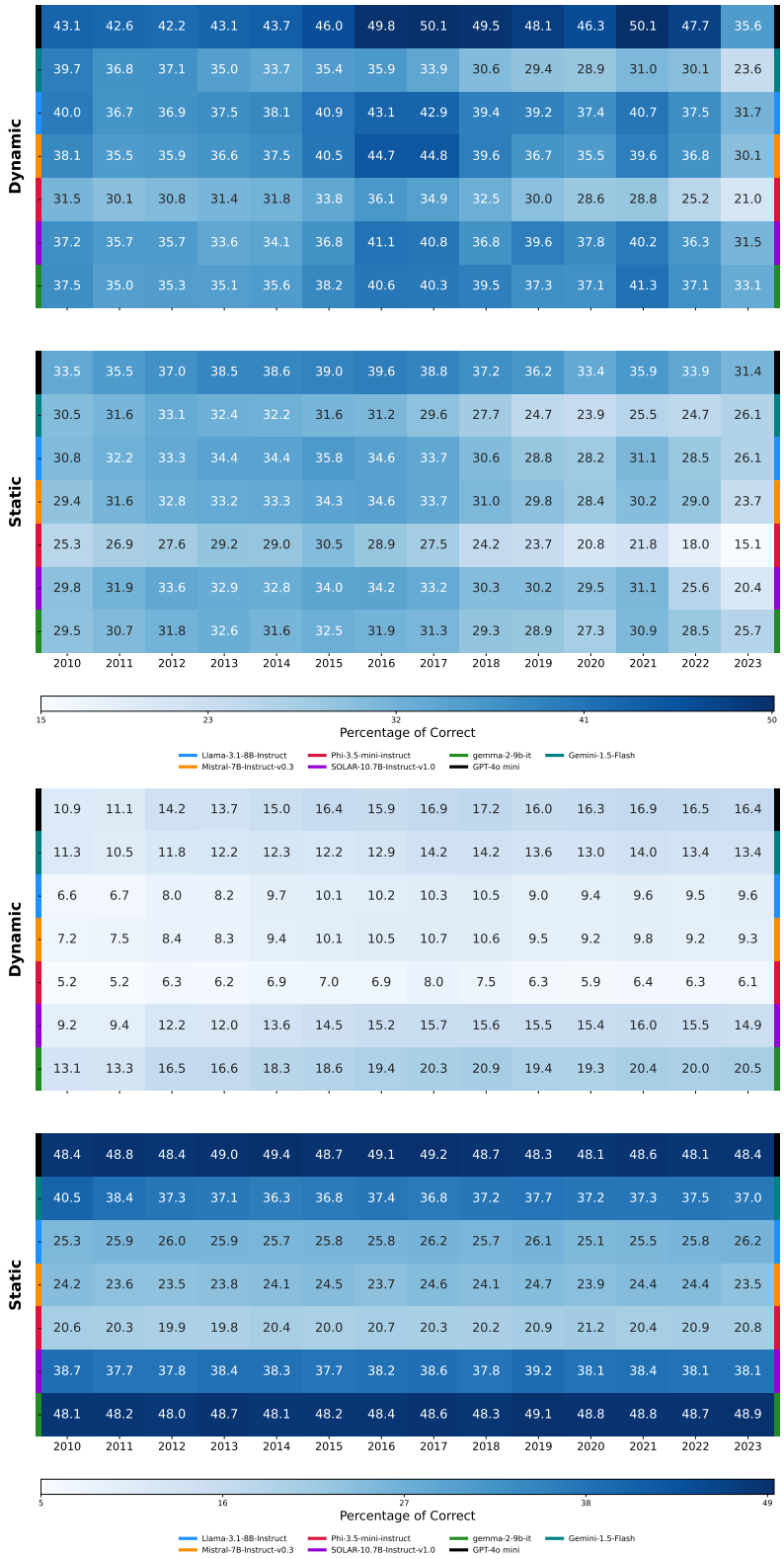

Figure 10: Total result of performance heatmap in general and legal domain

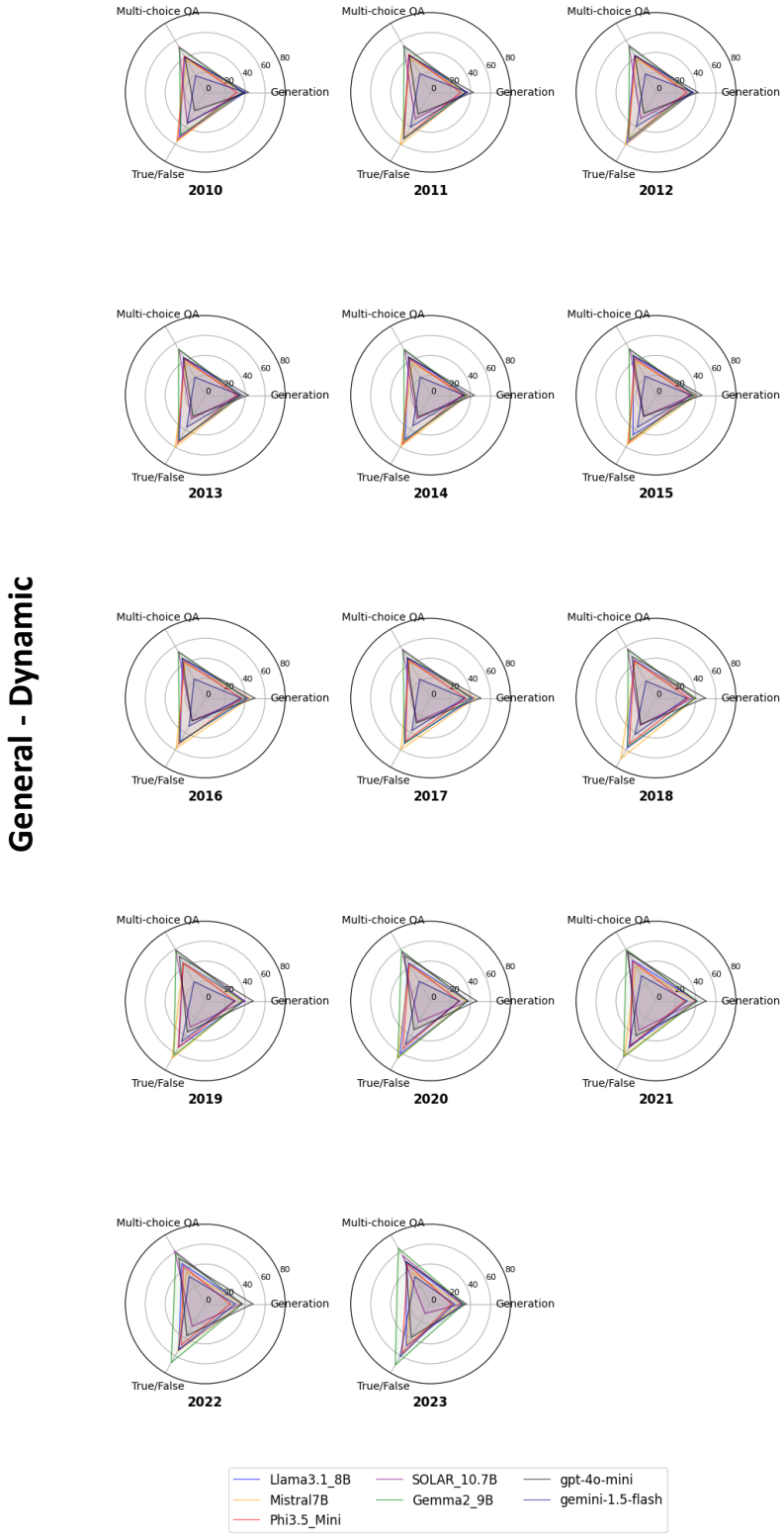

Figure 11: Total result of template-wise performance in general domain, dynamic dataset

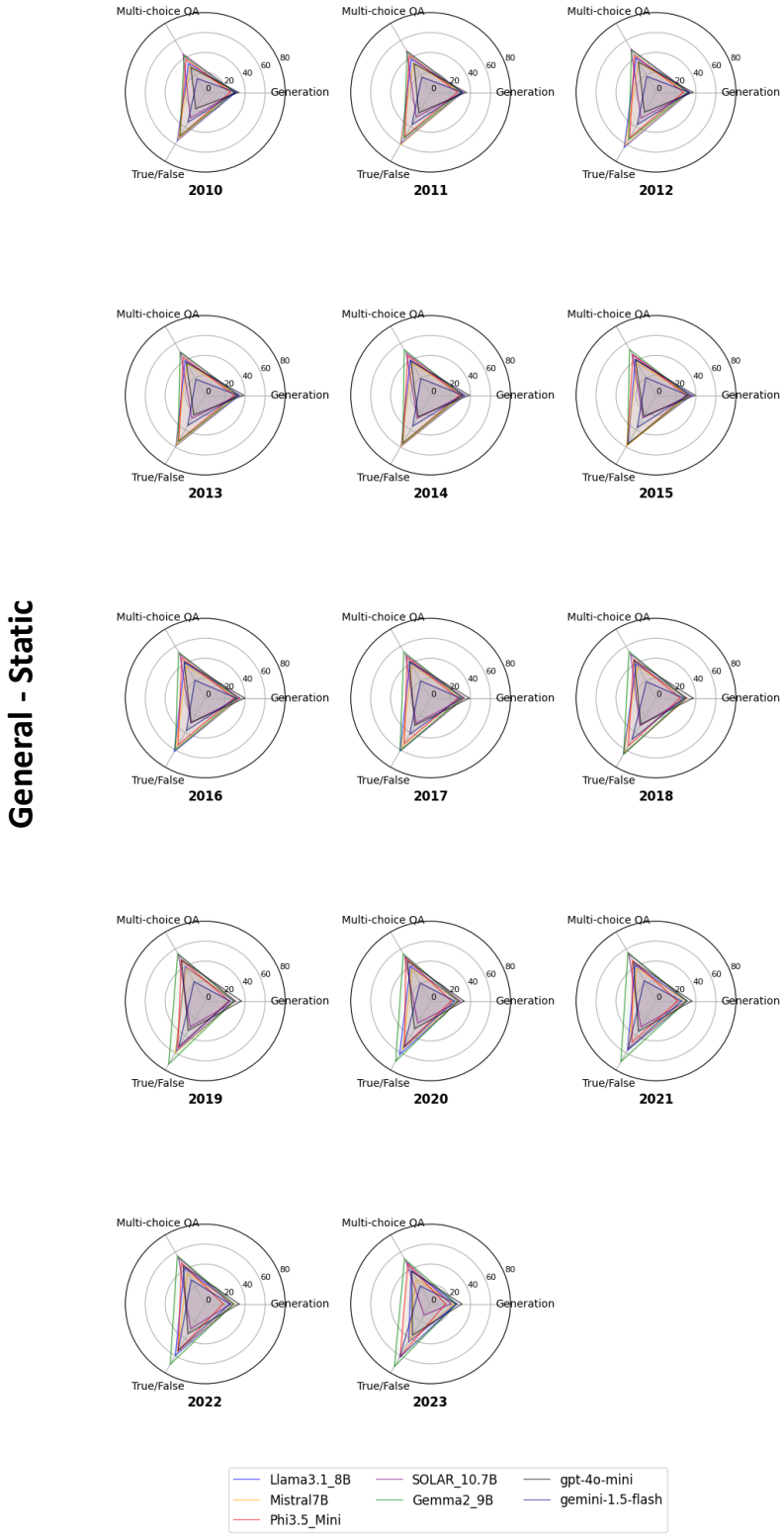

Figure 12: Total result of template-wise performance in general domain, static dataset

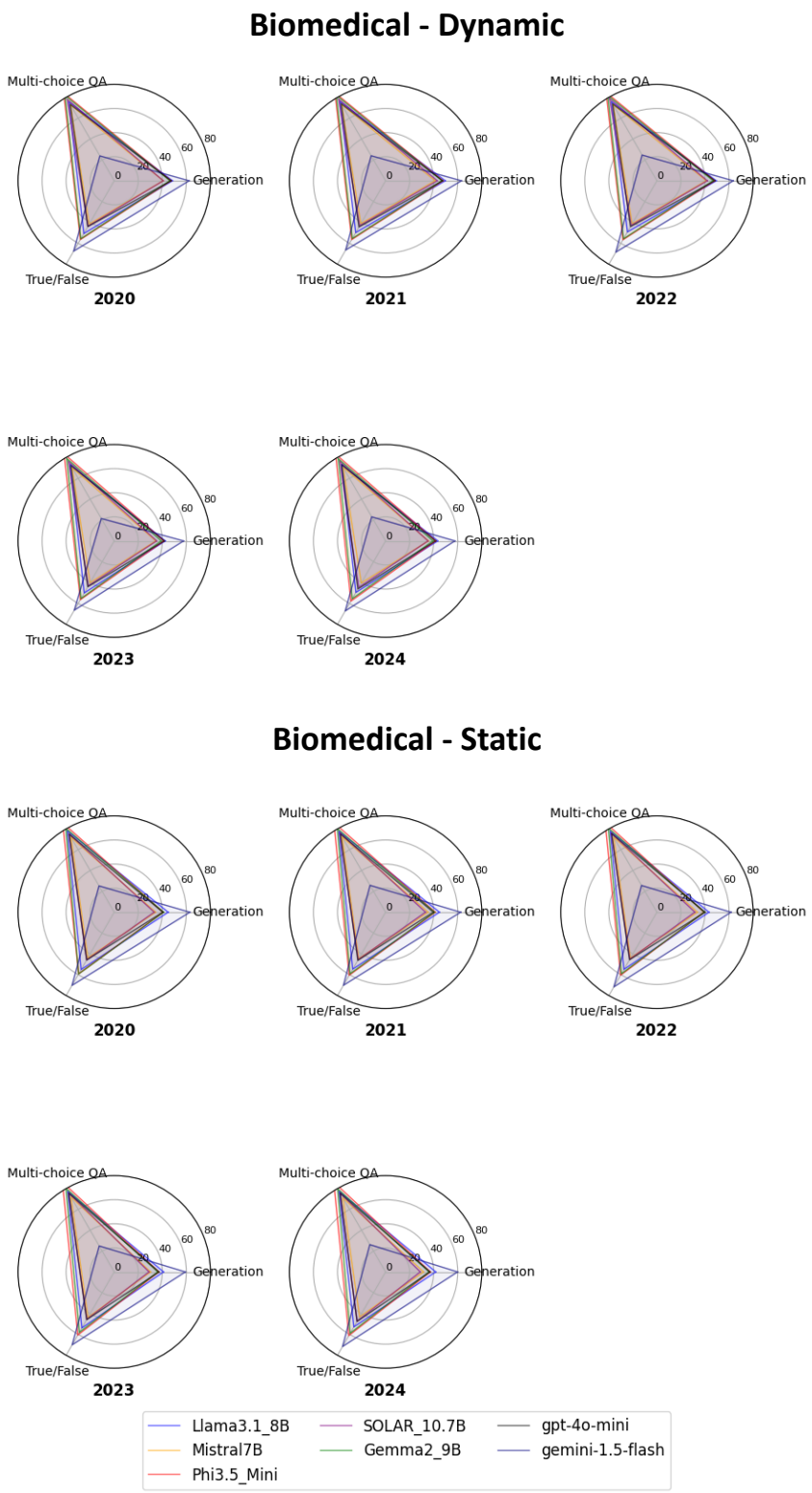

Figure 13: Total result of template-wise performance in biomedical domain, dynamic and static dataset

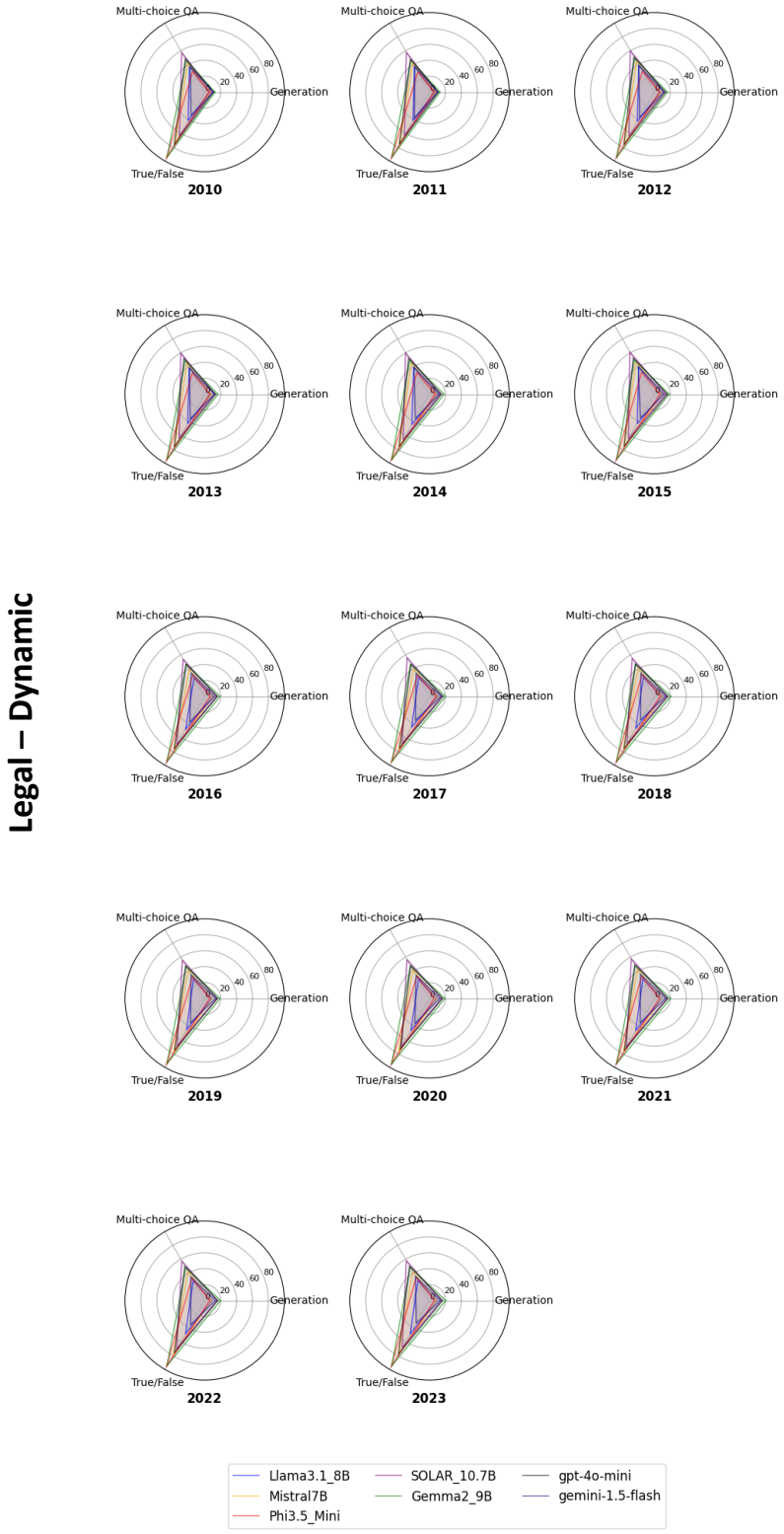

Figure 14: Total result of template-wise performance in legal domain, dynamic dataset

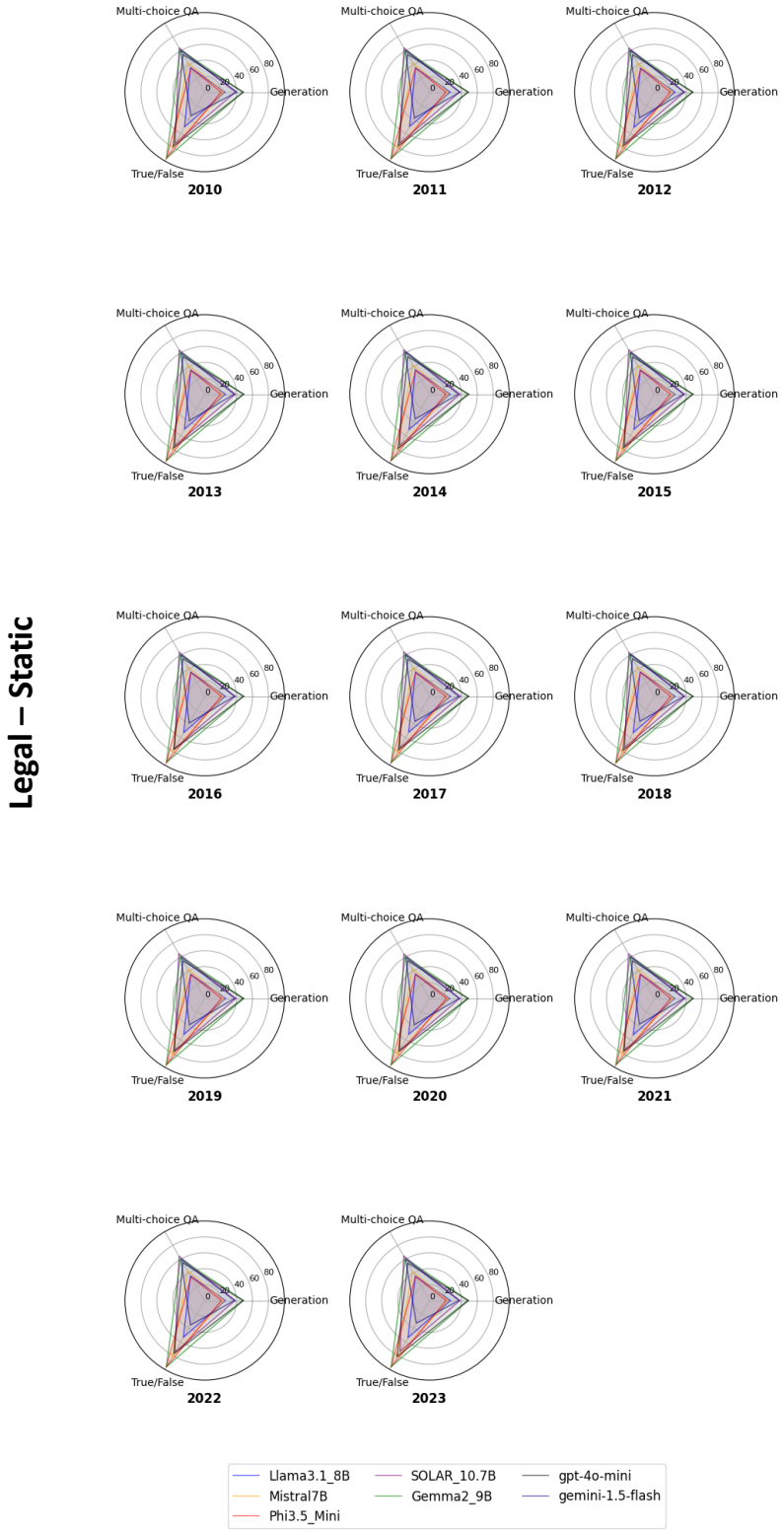

Figure 15: Total result of template-wise performance in legal domain, static dataset

Template: Generation

Q. In 2020, Pedro Braillard Poccard, position held, [Object]
A. member of the Argentine Chamber of Senators

Q. In 2020, Ana Santos Aramburo, position held, [Object]
A. Spain National Library general manager

Q. In 2020, James E. McPherson, position held, [Object]
A. United States Secretary of the Navy

Q. In 2020, Jesús Ávila de Grado, position held, [Object]
A. chief scientific officer

Q. In 2020, Donald Tusk, position held, [Object]
: (generate from here ...)

Template: MCQA

In 2020, what office does Pedro Braillard Poccard hold?
(a) member of the Argentine Chamber of Senators, (b) Minister of Foreign Affairs, (c) Governor of Corrientes Province, (d) Mayor of Buenos Aires

(a) member of the Argentine Chamber of Senators

In 2020, what office does Ana Santos Aramburo hold?
(a) Minister of Culture and Sports of Spain, (b) Director of the Prado Museum, (c) Spain National Library general manager, (d) President of the Spanish Royal Academy

(c) Spain National Library general manager

In 2020, what office does James E. McPherson hold?
(a) United States Secretary of Homeland Security, (b) United States Attorney General, (c) United States Secretary of the Navy, (d) United States Secretary of Defense

(c) United States Secretary of the Navy

In 2020, what office does Jesús Ávila de Grado hold?
(a) President of the National Research Council, (b) Minister of Health, (c) Director of the World Health Organization, (d) chief scientific officer

(d) chief scientific officer

In 2020, what office does Donald Tusk hold?
(a) President of the European Commission, (b) President of Poland, (c) Chancellor of Germany, (d) chairperson

: (generate from here ...)

Table 8: Example of two templates: *Generation* and *MCQA*. Domain: general, Subject: Donald Tusk, Relation: position held (P39), year to target: 2020.

Template: TF
Q. In 2020, Pedro Braillard Poccard, position held, member of the Argentine Chamber of Senators A. true Q. In 2020, Ana Santos Aramburo, position held, Director of the Prado Museum A. false Q. In 2020, James E. McPherson, position held, United States Secretary of Defense A. false Q. In 2020, Jesús Ávila de Grado, position held, chief scientific officer A. true Q. In 2020, Donald Tusk, position held, Prime Minister of Poland : (generate from here ...)

Table 9: Example of *TF* templates. Domain: general, Subject: Donald Tusk, Relation: position held (P39), year to target: 2020.

Domain: General

[System]
You are an expert in natural language processing and logic puzzles, skilled at generating plausible yet misleading distractor options that challenge users to distinguish between correct and incorrect answers. Pay special attention to questions that involve negative phrasing, such as those containing "not" or "which does not," to ensure that the distractors do not confuse users into overlooking the negative aspect of the question.

[User]
The question "What office does 'Mitt Romney' hold?" can be answered with 'United States senator'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Governor of Massachusetts
2. Secretary of State
3. Speaker of the House

[User]
The question "Which sports team is 'Yann MVila' a member of?" can be answered with 'Rubin Kazan', 'France national association football team', 'Sunderland A.F.C.', 'France national under-21 association football team', 'Inter Milan', 'Stade Rennais F.C.'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Paris Saint-Germain F.C.
2. Olympique Lyonnais
3. AS Monaco FC

[User]
The question "`[Q]`" can be answered with '`[O1]`', '`[O2]`', ... , '`[On]`'. Create three plausible incorrect distractors for this question.

[Assistant]
: (generate from here ...)

Table 10: 2-shot prompt for generating three distractors in general domain.

Domain: Biomedical

[System]
You are an expert in natural language processing and logic puzzles, skilled at generating plausible yet misleading distractor options that challenge users to distinguish between correct and incorrect answers. Pay special attention to questions that involve negative phrasing, such as those containing "not" or "which does not," to ensure that the distractors do not confuse users into overlooking the negative aspect of the question.

[User]
The question "What is not the primary anatomic site of 'Rhabdomyosarcoma of the orbit'?" can be answered with 'Bones set', 'structure bone', 'Bone structure', 'os', 'Skeletal bone', 'Bone structure (body structure)', 'bones structure', 'bone structures', 'Ossa', 'Set of bone organs', 'Bone organ', 'Skeleton system', 'Osseous', 'Bone', 'NOS', 'Bones', 'bones structures', 'Bone (organ)', 'Bone', 'skeletal', 'Set of bones', 'Bones', 'NOS'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Orbital connective tissue
2. Extrinsic ocular muscles
3. Orbital soft tissue

(two more shots)

[User]
The question "What negatively regulates 'Vitellogenesis'?" can be answered with 'downregulation of yolk production', 'downregulation of vitellogenesis', 'down regulation of vitellogenesis', 'inhibition of yolk production', 'negative regulation of vitellogenesis', 'down regulation of yolk production', 'down-regulation of yolk production', 'inhibition of vitellogenesis', 'down-regulation of vitellogenesis', 'negative regulation of yolk production'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Partial left salpingectomy
2. Unilateral oophorectomy
3. Hysterectomy

[User]
The question "`[Q]`" can be answered with '`[O1]`', '`[O2]`', ... , '`[On]`'. Create three plausible incorrect distractors for this question.

[Assistant]
: (generate from here ...)

Table 11: 4-shot prompt for generating three distractors in biomedical domain.

Domain: Legal

[System]
You are a legal expert skilled in crafting challenging fill-in-the-blank questions and generating plausible yet misleading distractor options. You will receive a question and answer where part of a legal text has been blanked out. For the provided question, create three plausible incorrect distractors that challenge users to distinguish between correct and incorrect answers.

[User]
The question
"""
... *(beginning of text omitted)*
Subpart B—Certification of Substantially Equivalent Agencies
Substantial equivalency certification is granted if the ____ determines that a state or local agency enforces a law that is substantially equivalent to the Fair Housing Act with regard to substantive rights, procedures, remedies, and the availability of judicial review. The Department has developed a two-phase process of substantial equivalency certification.
"""
can be answered with 'Department'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Secretary
2. Commission
3. Board

[User]
The question
"""
... *(beginning of text omitted)*
Subpart B—Minimum Standards for Substantial Compliance by States
Within the period defined in § 383.73(___ :
"""
can be answered with 'f) of this title, the State shall', 'h) of this subchapter, the State must'. Create three plausible incorrect distractors for this question.

[Assistant]
1. g) of this chapter, the State will
2. e) of this section, the State is required to
3. d) of this part, the State should

[User]
The question
"""
`[Q]`
"""
can be answered with '`[O1]`', '`[O2]`', ... , '`[On]`'. Create three plausible incorrect distractors for this question.

[Assistant]
: (generate from here ...)

Table 12: 2-shot prompt for generating three distractors in legal domain.

Domain: CommonSense

[System]
You are an expert in natural language processing and logic puzzles, skilled at generating plausible yet misleading distractor options that challenge users to distinguish between correct and incorrect answers. Pay special attention to questions that involve negative phrasing, such as those containing "not" or "which does not," to ensure that the distractors do not confuse users into overlooking the negative aspect of the question.

[User]
The question "What properties does 'human' not have?" can be answered with 'strong enough to lift horse', 'strong enough to lift truck', 'strong enough to lift car', 'strong enough to lift cow', 'strong enough to lift building'. Create three plausible incorrect distractors for this question.

[Assistant]
1. strong enough to lift a motorcycle
2. strong enough to lift a large suitcase
3. strong enough to lift a heavy stone

[User]
The question "What is synonym for 'salix tristis'?" can be answered with 'dwarf gray willow'. Create three plausible incorrect distractors for this question.

[Assistant]
1. green weeping willow
2. dwarf green willow
3. gray weeping willow

[User]
The question "What does 'kids' desire?" can be answered with 'play games', 'play', 'find crabs in tidal pools', 'read games magazines', 'play with', 'play with toys', 'learn things'. Create three plausible incorrect distractors for this question.

[Assistant]
1. eat vegetables
2. do homework
3. fold laundry

[User]
The question "What is 'oil and water' not capable of?" can be answered with 'mix'. Create three plausible incorrect distractors for this question.

[Assistant]
1. boil
2. freeze
3. emulsify

[User]
The question "`[Q]`" can be answered with '`[O1]`', '`[O2]`', ... , '`[On]`'. Create three plausible incorrect distractors for this question.

[Assistant]
: (generate from here ...)

Table 13: 4-shot prompt for generating three distractors in commonsense domain.

Prompt: Mathematics / Data Structure & Algorithm

[System]
You are an expert in mathematics and computer science, skilled at generating plausible yet misleading distractor options that challenge users to distinguish between correct and incorrect answers. Pay special attention to questions that involve negative phrasing, such as those containing "not" or "which does not," to ensure that the distractors do not confuse users into overlooking the negative aspect of the question.

[User]
The question "Which of the following is unrelated to 'Output'?" can be answered with 'Left child', 'Set', 'Depth', 'Modify', 'Post-order traversal', 'Dictionary', 'Predecessor', 'Array', 'Delete', 'Sparse matrix', 'Infix expression', 'Leaf node', 'Shortest path', 'Right subtree', 'Node', 'In-order traversal', 'Hashing', 'Head node', 'Pointer', 'Level-order traversal', 'Keyword', 'Pattern string', 'Record', 'Determinism', 'Linked list', 'Critical Path', 'Function name', 'Connected component', 'Loop statement', 'Robustness', 'Inverted index', 'Preorder traversal', 'Memory', 'Dequeue', 'Singly Linked List', 'Image'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Print statement
2. Display buffer
3. Output stream

[User]
The question "What is synonym for 'Ancestor'?" can be answered with 'Parent'. Create three plausible incorrect distractors for this question.

[Assistant]
1. Descendant
2. Sibling
3. Offspring

[User]
The question "`[Q]`" can be answered with '`[O1]`', '`[O2]`', ... , '`[On]`'. Create three plausible incorrect distractors for this question.

[Assistant]
: (generate from here ...)

Table 14: 2-shot prompt for generating three distractors in mathematics, data structure, and algorithm domain.