DyKnow: Dynamically Verifying Time-Sensitive Factual Knowledge in LLMs

Seyed Mahed Mousavi, Simone Alghisi, Giuseppe Riccardi
Signals and Interactive Systems Lab, University of Trento, Italy
{mahed.mousavi,s.alghisi,giuseppe.riccardi}@unitn.it

Abstract

LLMs acquire knowledge from massive data snapshots collected at different timestamps. Their knowledge is then commonly evaluated using static benchmarks. However, factual knowledge is generally subject to time-sensitive changes, and static benchmarks cannot address those cases. We present an approach to dynamically evaluate the knowledge in LLMs and their time-sensitiveness against Wikidata, a publicly available up-to-date knowledge graph. We evaluate the time-sensitive knowledge in twenty-four private and open-source LLMs, as well as the effectiveness of four editing methods in updating the outdated facts. Our results show that 1) outdatedness is a critical problem across state-of-the-art LLMs; 2) LLMs output inconsistent answers when prompted with slight variations of the question prompt; and 3) the performance of the state-of-the-art knowledge editing algorithms is very limited, as they can not reduce the cases of outdatedness and output inconsistency.

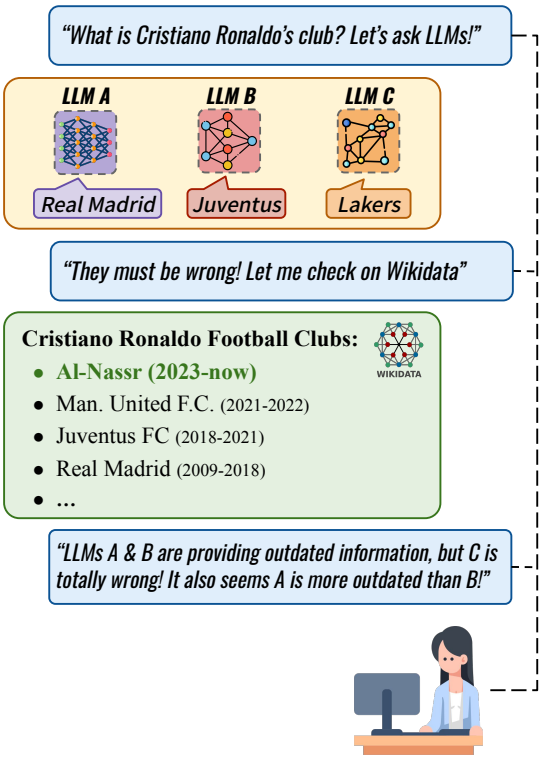

Figure 1: LLMs *A*, *B*, and *C* may respond with outdated (*Real Madrid, Juventus*) and irrelevant (*Lakers*) responses, respectively, to the user question:"*What is Cristiano Ronaldo's club?*". Wikidata contains up-to-date information to assess the models' accuracy and time-sensitiveness.

1 Introduction

Large Language Models (LLMs) have been compared to traditional *knowledge repositories*, such as knowledge bases, knowledge graphs, and search engines regarding their capability to retrieve factual knowledge (Cohen et al., 2023b; Sun et al., 2023; Pinter and Elhadad, 2023; Hu et al., 2024). A critical requirement for a reliable knowledge repository is to maintain the accuracy of the factual information it contains. The factual knowledge has a dynamic nature and can change significantly from what has been first inserted; and in the case of LLMs what was observed during the training stage.

LLMs are static models that are prone to generating invalid and contradicting information, and eventually getting outdated over time. They derive their knowledge from vast and often unoptimized collections of data snapshots, that are collected at different timestamps (Dhingra et al., 2022), and typically contain substantial overlaps (Soldaini et al., 2024). These collections contain factual associations interspersed with inaccuracies, outdated information, and contradictions.

The maintenance of LLMs' knowledge requires a systematic approach to *i*) identifying the outdated knowledge, *ii*) locating it within the model parameters, and *iii*) applying the necessary changes. There have been interesting studies on locating the factual associations in LLMs (Meng et al., 2022a), understanding how they are retrieved (Geva et al., 2023), and editing them (Li et al., 2024). However, there are no studies on detecting outdated knowledge in

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 8014–8029
November 12-16, 2024

LLMs (*i*). While several static benchmarks have been proposed to assess the factuality of LLMs (Hu et al., 2024), such benchmarks are not suitable for detecting outdated knowledge in LLMs. Due to the dynamic nature of knowledge, a static benchmark can quickly become outdated and lose its relevance. Moreover, it can be prone to leakage into the training data of future models (contamination). Consecutively, studies on editing techniques are mainly based on annotated edit-target datasets of synthetically generated counterfacts, leaving a gap in understanding how these methods perform with real-world data across diverse domains (Zhang et al., 2023).

To address the issues of static benchmarks, dynamic benchmarking has been proposed where the data points are continuously updated to reflect real-time scenarios. Despite extensive research on static benchmarking, theoretical and empirical research on dynamic benchmarking is very limited (Shirali et al., 2023), making it challenging and expensive to construct a valid dynamic benchmark (Yin et al., 2023). We present an approach to dynamically benchmark the factual knowledge in LLMs using Wikidata[1].

For each factual association in the form of *(subject, property, attribute)*, the most current attribute values are obtained from the Wikidata knowledge base at the time of evaluation, in addition to the complete list of outdated values along with their validity interval (for example, in Figure 1, the validity interval of "`Juventus FC`" as the correct attribute for "`Cristiano Ronaldo's current football club`" is 2018-2021). The attribute value generated by the model is then validated against this comprehensive list, evaluating the accuracy and timeliness of the model responses.

We assess the efficacy of the proposed approach by investigating the following Research Questions (RQs):

- ***RQ1. How reliable are state-of-the-art LLMs in responding to time-sensitive factual questions?*** We evaluate the knowledge of 24 LLMs regarding a diverse set of time-sensitive facts. We further evaluate the consistency of the model outputs across various prompts, as an indicator of input-bound uncertainty (Portillo Wightman et al., 2023; Lyu et al., 2024).
- ***RQ2. Can we estimate the temporal interval of the data used to (pre-)train the LLMs?*** We analyze the outputs of each model based on their validity intervals and approximate the temporal interval of the (pre-)training data. We compare our estimations with the reports from models that have disclosed details of their (pre-)training data.
- ***RQ3. Can knowledge editing methods improve the accuracy and consistency of LLMs regarding real-world time-sensitive facts?*** We select four outdated LLMs and apply four editing algorithms to update their outdated knowledge regarding the real world. We evaluate the effectiveness and scalability of the editing algorithms in updating LLMs regarding real-world facts.

[1]Link to our Repository

2 Literature Review

LLMs as Knowledge Repositories Pinter and Elhadad (2023) noted that current LLMs fall short as knowledge repositories due to issues with *editing*, *logical consistency*, *reasoning*, and *interoperability*. They identified problems with existing knowledge editing techniques, such as catastrophic forgetting (Ratcliff, 1990), limitations on the number of edits (Mitchell et al., 2021), ripple effect failures (Cohen et al., 2023a), and lack of robustness (Brown et al., 2023; Hase et al., 2023). Mazzia et al. (2023) summarized model editing research across computer vision and NLP fields. Zhang et al. (2023) studied methods for aligning LLMs with real-world knowledge, pointing out issues such as unrealistic evaluation settings, synthetic datasets, insufficient quantitative analysis, and lack of studies on detecting outdated knowledge in LLMs.

Knowledge Benchmarks Studies on temporal reasoning in LLMs evaluate the knowledge of the model regarding a specific time in the past via an explicit time-specifier (Chen et al., 2021; Gupta et al., 2023), or in more challenging settings multiple temporal factors (Wei et al., 2023). Yu et al. (2023) proposed an evaluation setup assessing models on memorization, understanding, application, and creation of knowledge. Yin et al. (2023) discussed challenges in building dynamic factual benchmarks and suggested generating artificial new knowledge by randomly altering entities/relations within the same ontological class. While the mentioned studies presented static benchmarks, Kasai et al. (2022) introduced RealTime QA, a benchmark with 30 weekly questions and answers for LLM evaluation. Meanwhile, Jang et al. (2022) presented an

approach to track changes in knowledge by comparing consecutive snapshots of Wikipedia and retraining the models on the identified differences.

3 DyKnow 🦕: Dynamic Knowledge Validation

The benchmark for evaluating time-sensitive knowledge in LLMs must be model-agnostic and long-lasting since it must not become outdated as the models. A static benchmark lacks these characteristics, cannot capture the changing world, and can lead to data contamination.

We present a cost-effective approach to dynamically benchmark the factual knowledge in LLMs using Wikidata. Wikidata is a multilingual knowledge graph that is continuously and collaboratively edited to maintain up-to-date information (Vrandečić and Krötzsch, 2014). Factual Knowledge in Wikidata is presented by *properties* that connect *subject* nodes to *attribute* values. For example, "Cristiano Ronaldo's current football club" factual knowledge is presented by the property `"member of sports team"` that connects the subject `"Cristiano Ronaldo"` to the current attribute value, at the time of paper `"Al-Nassr"`. Furthermore, the *attribute* values in Wikidata are accompanied by *qualifiers*, which provide additional context and specificity regarding the attribute values such as geographical locations, measurement units, as well as start and end dates for attribute values that have a temporal validity interval. For instance, the attribute value `"Al-Nassr"` is accompanied by start and end date quantifiers `"2023-Now"`. Besides the current attribute value for each factual triplet, Wikidata maintains all the previously correct attribute values in addition to their corresponding start and end date quantifiers, indicating the corresponding temporal validity intervals. Therefore, the complete list of attributes for "Cristiano Ronaldo's current football club" factual knowledge consists of [`Al-Nassr` $_{\texttt{2023-Now}}$, `Manchester United F.C.` $_{\texttt{2021-2022}}$, `Juventus FC` $_{\texttt{2018-2021}}$, `Real Madrid` $_{\texttt{2009-2018}}$, ...] (Figure 1).

Instead of relying on static ground truth values, we evaluate the models' outputs with the list of attribute values retrieved dynamically from the Wikidata knowledge base at the time of evaluation. We assess the knowledge of the model regarding each fact as:

- **Correct** when the model outputs the most up-to-date value from the list; we further categorize the *Incorrect* outputs of the model as
- **Outdated** when the model outputs a value that is not correct anymore and now is outdated; and
- **Irrelevant** when the model output is not present in the Wikidata list (e.g. due to hallucination or contradicting/false information in the training data)

Furthermore, by analyzing the correct and outdated outputs of each model according to their validity interval, we can approximate the temporal interval of the data used for (pre-)training the models. For instance, if a model provides outdated responses to time-sensitive questions, with the oldest responses dating back to 2016 and the most recent ones correct until 2019, we can infer that the model was likely trained on data collected up to 2020, encompassing documents from 2016 to 2019.

4 Validating DyKnow 🦕

To assess the efficacy of the proposed approach, we evaluate 24 LLMs on 130 time-sensitive facts including countries' politicians, athletes' clubs, and organizations' roles. This allow us to introduce diversity in the dataset by having human subjects, organization subjects, and country subjects with a diverse set of properties to query the models.

Time-Sensitive Facts We aim to select subject entities that are most likely to be frequently present in the training data of most LLMs. This choice is motivated by studies showing the performance of LLMs regarding factual information about an entity depends on its frequency in the training data (Pinter and Elhadad, 2023; Mallen et al., 2023). We select the top 50 countries by Gross Domestic Product (GDP) in 2023, the top 30 athletes of 2023 (10 soccer players, 10 basketball players, and 10 Formula 1 drivers), and 25 public and private organizations (the top 20 companies by revenue and the top 5 organizations by influence). For each country, we query the models about the "*head of state*" (e.g., president, king) and the "*head of government*" (e.g., prime minister, premier). For each athlete, we query about their *sports team*, and for each organization, we ask about the corresponding *directorial role* (e.g., CEO, chairperson). After manually removing the subjects with missing property/attributes in Wikidata, the final list of time-sensitive facts to evaluate the LLMs consists of 78

facts about 47 countries, 28 facts about 28 athletes, and 24 facts about 23 organizations. The complete list of subject entities and properties as time-sensitive facts used in benchmarking the LLMs is presented in § Table 3.

LLMs We evaluate the following 24 LLMs: GPT-2 XL (Radford et al.), GPT-3[2] (Brown et al., 2020), T5 (3B) (Raffel et al., 2020), GPT-J (6B) (Wang and Komatsuzaki, 2021), ChatGPT (GPT-3.5)[3], Bloom (7B) (Workshop et al., 2022), Flan-T5 XL (Chung et al., 2022), GPT-4[4], Llama-2 (7B) & Llama-2 Chat (7B) (Touvron et al., 2023), Falcon (7B) & Falcon Instruct (7B) (Almazrouei et al., 2023), Vicuna v1.5 (7B) (Chiang et al., 2023), Mistral v0.1 (7B) & Mistral Instruct v0.1 (7B) (Jiang et al., 2023), Mixtral 8x7B v0.1 & Mixtral 8x7B Instruct v0.1 (Jiang et al., 2024), OLMo (1B & 7B) (Groeneveld et al., 2024), Llama-3 (8B) and Llama-3 Instruct (8B)[5], OpenELM (270M & 1.1B & 3B) (Mehta et al., 2024).

RQ1: LLMs' Time-Sensitive Knowledge

A. Knowledge Evaluation

Using DyKnow, we evaluate 24 LLMs regarding 130 time-sensitive facts about frequent subject entities in different categories (human subjects, organization subjects, and country subjects). For each time-sensitive fact, the outputs of the models are validated against a list of attribute values dynamically retrieved from Wikidata, classifying the outputs as **C**orrect, **O**utdated, and **I**rrelevant.

Prompting Strategy We develop a prompt template for each time-sensitive fact and subject group, including placeholders for subject names and, for countries, official titles. Using GPT-4, we generate four rephrased versions of each prompt as slightly perturbed lexicalizations and ask three human judges (researchers in our group) to review and validate the generated prompts. After collecting feedback and manual controls, three question prompts are selected for each fact. We then queried the models for each time-sensitive fact using the selected three prompts. In contrast to studies on the temporal reasoning of LLMs (Chen et al., 2021; Wei et al., 2023; Gupta et al., 2023), our questions are framed in the present tense, omit explicit time specifiers, and seek the *currently correct* answer.

[2]davinci-002
[3]gpt-3.5-turbo-1106
[4]gpt-4-1106-preview
[5]LLaMA-3

(Year) Model	Correct	Outdated	Irrelevant
(2019) GPT-2	26%	42%	32%
(2020) GPT-3	42%	47%	12%
(2020) T5	11%	21%	68%
(2021) GPT-J	41%	46%	13%
(2022) Bloom	35%	49%	16%
(2022) Flan-T5	18%	39%	43%
(2023) Llama-2	51%	42%	7%
(2023) Falcon	42%	47%	11%
(2023) Mistral	53%	39%	8%
(2023) Mixtral	48%	42%	10%
(2024) OLMo 1B	37%	40%	23%
(2024) OLMo 7B	35%	36%	29%
(2024) Llama-3	57%	36%	7%
(2024) OpenELM 270M	12%	28%	61%
(2024) OpenELM 1.1B	35%	47%	18%
(2024) OpenELM 3B	42%	42%	16%
(2022) ChatGPT	57%	35%	8%
(2023) GPT-4	80%	13%	7%
(2023) Llama-2$_{C.}$	51%	37%	12%
(2023) Falcon$_{I.}$	44%	41%	15%
(2023) Vicuna	52%	33%	15%
(2023) Mistral$_{I.}$	52%	32%	16%
(2023) Mixtral$_{I.}$	62%	29%	9%
(2024) Llama-3$_{I.}$	76%	14%	10%

Table 1: Benchmarking 24 LLMs with time-sensitive knowledge via *Upper Bound*. The table presents the percentage of **C**orrect answers that are valid and up-to-date, **O**utdated answers that are not valid anymore, and **I**rrelevant outputs. Models below the dashed line were prompted with an additional prefix "Answer with the name only". Subscripts $I.$ and $C.$ stand for *Instruct* and *Chat*, respectively.

§ Table 3 presents the prompt templates used to query the models for time-sensitive facts for each subject category.

Upper Bound We validate the generated outputs using an "Upper Bound" approach. If the model provides the correct (up-to-date) answer to at least one of the three prompts, we consider it a success, indicating that the information in the model regarding that specific fact is current. If the model fails to give a correct answer but provides an outdated response to at least one of the prompts, we classify the information in the model as outdated. Irrelevant outputs may occur due to several reasons: a) the model may not have learned the specific time-sensitive fact during (pre-)training or fine-tuning, b) hallucinations or conflicting/false information in

the training data, or c) the information may not be retrievable using our prompts.

Results Table 1 shows the results of this evaluation on 24 LLMs[6]. The results highlight concerning issues regarding the currency of the models' knowledge about frequent subject entities. Even the best-performing models exhibit non-negligible percentages of outdated and irrelevant answers, which can be problematic in real-world applications where up-to-date and accurate information is crucial. GPT-4 (2023) demonstrates a high rate of correct responses, while 20% of its outputs are either outdated or irrelevant. Similarly, more recent models such as Llama-3 (2024), OLMo (2024), and OpenELM (2024) output incorrect (outdated and irrelevant) responses to more than 40% of the questions. As expected, older models like GPT-2 (2019) and GPT-3 (2020) demonstrate lower levels of up-to-dateness. These statistics imply that a significant portion of the models' outputs are either outdated or irrelevant, potentially leading to misinformation if relied upon.

B. Output Consistency

The consistency of model outputs across various prompts, known as *prompt agreement*, has been examined in the literature as an indicator of input-bound uncertainty (Portillo Wightman et al., 2023; Lyu et al., 2024). These studies are based on the premise that higher consistency across different prompts signals lower uncertainty in the model prediction. By querying the LLMs for each time-sensitive fact using the selected three prompts (§ Table 3), we observe that they often generate inconsistent answers to slightly modified versions of the same prompt.

Results Figure 2 presents the prompt agreement level, i.e. the consistency of outputs across different prompts for all models (the agreement percentage for each model in presented in § Table 5). The results show that prompt agreement varies significantly across different models, with most LLMs demonstrating low levels of prompt agreement, indicating that they produce varying responses to slightly altered versions of the same question. There is a trend of improvement in prompt agreement over time, with more recent models showing higher consistency in their responses. Furthermore, instruction-tuned models demonstrate a comparatively higher prompt agreement. ChatGPT (2022) and GPT-4 (2023) exhibit the highest prompt agreement. Other high performers include Mistral$_{I.}$ (2023), Mixtral$_{I.}$ (2023), and Llama-3$_{I.}$ (2024). In contrast, OpenELM 270M (2024) has the lowest agreement among the models. These results highlight the high sensitivity in the auto-regressive generation process can lead to different, incorrect, or irrelevant outputs.

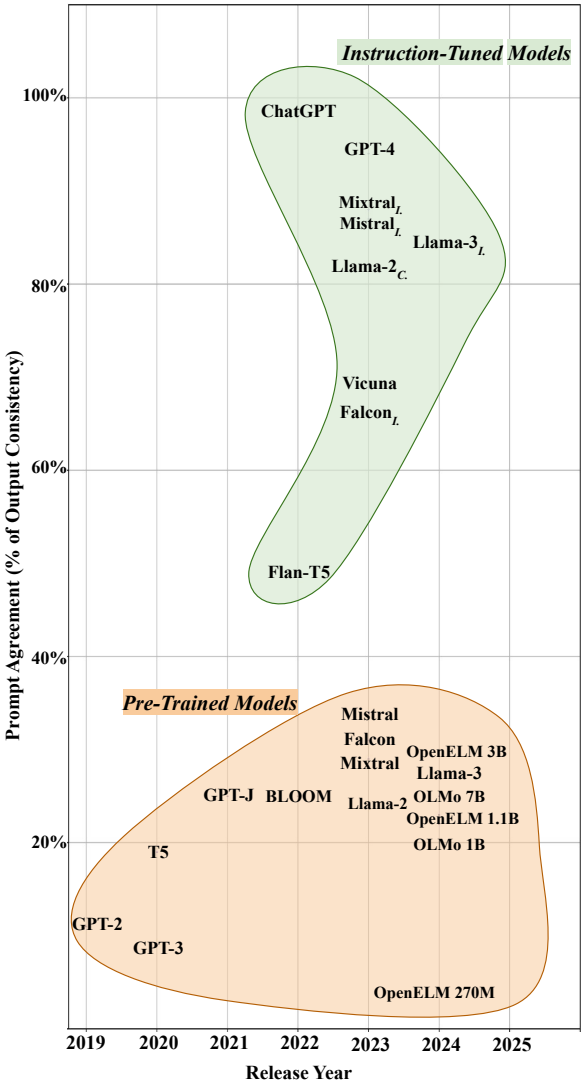

Figure 2: The level of prompt agreement for each model across three prompts for each time-sensitive question. Subscripts $I.$ and $C.$ stand for *Instruct* and *Chat*, respectively. Instruction-tuned models demonstrate a comparatively higher prompt agreement.

RQ2: LLMs' Data Interval Approximation

Each attribute value in Wikidata is accompanied by start and end date quantifiers, indicating the corresponding temporal validity intervals. We analyze the correct and outdated outputs of each model

[6]The models are evaluated with the answer sets retrieved on 18 December 2023.

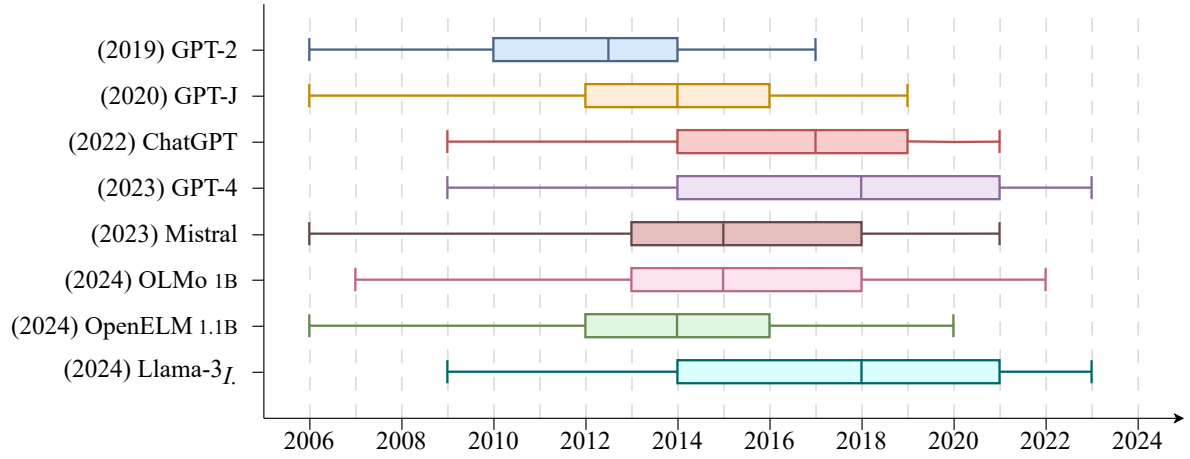

Figure 3: Approximating the temporal interval of the data used for (pre-)training LLMs following our evaluation regarding time-sensitive knowledge. The y-axis presents the evaluated LLMs with their release year in parentheses. The box plots present the distribution of the generated responses for each LLM according to their validity interval. For instance, the responses of OpenELM 1.1B range from 2006 to 2020, with a concentrated period between 2012 and 2016, suggesting that the mode is trained on comparatively older datasets.

according to their temporal validity intervals and approximate the temporal interval of the data used for (pre-)training the models.

Results The results of this analysis for GPT-{2,J,4}, ChatGPT, Mistral, OLMo 1B, OpenELM 1.1B, and Llama-3$_{I.}$ are presented in Figure 3 (The results for the remaining LLMs are presented in § Figures 5 and 6). Regarding the GPT model family, we approximate that older models such as GPT-{2,J} are trained on older datasets as a considerable portion of their responses date back to before 2009, contributing to their outdatedness compared to relatively new models, i.e. ChatGPT, GPT-4. There is a trend of improvement in the GPT family over time, as each model demonstrates a more recent median and maximum date compared to preceding models. While the maximum data value for ChatGPT is 2021, GPT-4 has generated responses with information from 2022 and 2023. This finding aligns with the OpenAI API report, which states that the training data for ChatGPT includes information "up to September 2021", while the training data for GPT-4 includes information "up to April 2023" [7]. Regarding recently released models, OLMo 1B generates a broad range of responses from 2006 to 2022 with the central part of the data from 2013 to 2018. This finding suggests that the model is (pre-)trained on a wide span of data and is in line with OLMo 1B data sheet paper (Soldaini et al., 2024). Llama-3$_{I.}$ demonstrates the same temporal distribution as GPT-4. Instead, the responses of OpenELM 1.1B range from 2006 to 2020, with a concentrated period between 2012 and 2016, suggesting that the model is trained on comparatively older datasets. In general, this analysis indicates that more recent models tend to include data from the last few years, leading to potentially more correct outputs. However, the presence of outdated responses in models highlights the importance of regular updates to maintain the currency and accuracy of the (pre-)training data and the models.

[7] OpenAI API Link

RQ3: Updating LLMs' Knowledge

Studies on editing techniques primarily rely on annotated edit-target datasets of synthetically generated counterfacts, leaving a gap in understanding their performance with real-world data across diverse domains (Zhang et al., 2023). To bridge this gap, we select four outdated LLMs and evaluate the efficacy of four editing algorithms to update their outdated knowledge on real-world data. Regarding the LLMs, we have selected GPT-{2,J} due to generating a high percentage of outdated responses among models in Table 1; and Llama-2$_{C.}$ and Mistral$_{I.}$ since, despite being relatively new models, provide outdated information to around 30% of the questions.

Methods Regarding algorithms that modify LLM parameters to incorporate edited knowledge, we evaluate two methods. First, **ROME** (Meng et al., 2022a) locates relevant parameters in the feed-forward layers and inserts new key-value associations as a least squares problem with a linear

Model	#Outdated Facts	Knowledge Editing			
		Modifying Parameters		Preserving Parameters	
		ROME	MEMIT	SERAC	IKE
(2019) GPT-2	54	17%	33%	4%	49%
(2021) GPT-J	60	11%	83%	0%	97%‡
(2023) Llama-$2_{C.}$	48	4%	77%	36%	18%
(2023) Mistral$_{I.}$	41	0%	0%	—	92%‡

Table 2: Performance of different knowledge editing methods for updating the outdated facts in LLMs, by the *harmonic mean* of efficacy success and paraphrase success (Meng et al., 2022a,b). ‡ indicates successful alignment of more than 85% outdated knowledge.

equality constraint. Second, **MEMIT** (Meng et al., 2022b) extends ROME to apply multiple edits simultaneously by operating on several layers in a single intervention. Regarding editing methods that preserve the original LLM parameters, we evaluate two approaches. First, **SERAC** (Mitchell et al., 2022) uses external memory to store new facts and a classifier to match question prompts with these stored facts. Depending on whether a match is found, the classifier decides whether to condition the generation of the model on the retrieved fact or not. Second, **IKE** (Zheng et al., 2023) utilizes in-context learning by constructing a prompt with the question, the corresponding up-to-date fact, and a context segment consisting of examples for answering the question. The constructed prompt is then presented to the model to generate an answer. Note that this method is not entirely realistic, as it requires relevant and up-to-date facts to be provided for each question. Further implementation details of the evaluated techniques are presented in § A.1.

Harmonic Mean We evaluate the methods using the harmonic mean of efficacy success and paraphrase success (Meng et al., 2022a,b). Efficacy success measures the proportion of correctly edited responses to the original question prompts (RQ1 A.), while paraphrase success assesses the model's performance on paraphrased versions of the prompts, serving as a generalization metric (RQ1 B.).

Results The results, presented in Table 2, indicate that the performance of editing methods is model-dependent (§ Table 6 reports the performance of the methods by paraphrasing success). Among the edited LLMs, GPT-J is a better candidate for updating as it achieves a high success rate by two methods, MEMIT and IKE. Among the methods that modify the LLM parameters, ROME demonstrates an overall poor performance and MEMIT significantly outperforms ROME in most

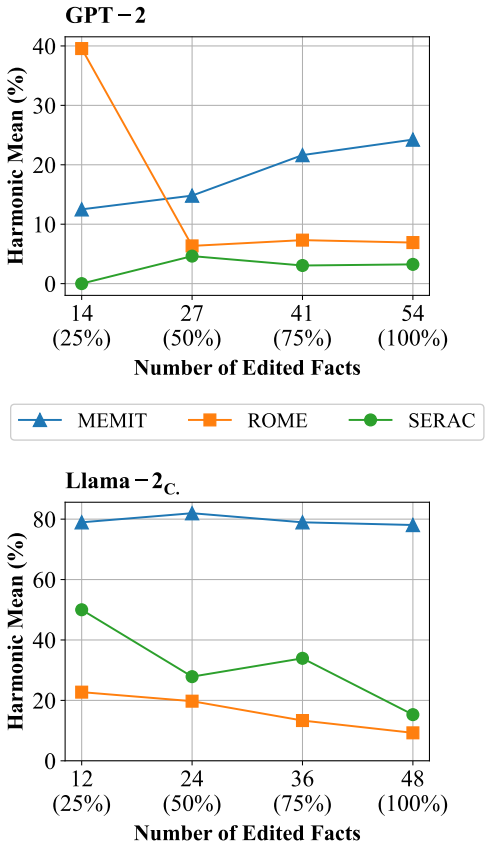

Figure 4: The scalability of editing algorithms for "*updating*" the outdated facts in GPT-2 and Llama-$2_{C.}$. The x-axis and y-axis represent the number of edits (in parenthesis the percentage of the total edits) and the harmonic mean of the models, respectively.

cases, especially for GPT-J and Llama-$2_{C.}$. However, none of these approaches apply to Mistral$_{I.}$ as it fails to output any meaningful sequence and generates only special tokens after parameter modification. Regarding editing methods that preserve the LLM parameters, SERAC fails to achieve high performance with updating at best less than 40% of the outdated information in Llama-$2_{C.}$. Meanwhile, IKE achieves a high performance on GPT-J

and Mistral$_{I.}$, achieving over 90% success rates, indicating the effectiveness of in-context learning for these two models. In general, MEMIT and IKE are the standout methods for modifying and preserving parameters, respectively. MEMIT excels with GPT-J and Llama-2$_{C.}$, while IKE shows high success rates with GPT-J and Mistral$_{I.}$.

Scalability Studies We investigate the scalability of ROME, MEMIT, and SERAC in performing different subsets of edits on real-world facts in GPT-2 and Llama-2$_{C.}$. The results, shown in Figure 4, indicate that ROME exhibits a significant decline in performance for both models as the number of edited facts increases. In contrast, MEMIT demonstrates a more stable performance, with gradual improvement on GPT-2 as the number of edits increases. SERAC, meanwhile, maintains consistently low performance on GPT-2 and shows a decline on Llama-2$_{C.}$ as the number of edits rises. Overall, ROME and SERAC exhibit poor scalability and effectiveness in handling multiple edits, while MEMIT presents stable performance across increasing numbers of edits on both models.

5 Discussion

In this section, we discuss our findings in relation to the introduced research, as well as avenues for future work.

RQ1. How reliable are state-of-the-art LLMs in responding to time-sensitive factual questions? While recent models like GPT-4 and Llama-3$_{I.}$ show better performance than other models, the persistent presence of outdated and incorrect information across all models suggests that current LLMs are still far from reliable knowledge sources. Furthermore, the high sensitivity of the auto-regressive generation process to slight variations in question lexicalization can lead to contradicting and sometimes incorrect or irrelevant outputs. This unreliability underscores the critical need for further refinement in training methodologies and updating mechanisms to consistently ensure these models provide accurate information at any time.

RQ2. Can we estimate the temporal interval of the data used to (pre-)train the LLMs? Our approximations align with the models' reports that disclose the data used during (pre-)training. This analysis indicates that comparatively recent models tend to include data from the last few years, leading to potentially more correct outputs. However, the presence of outdated facts in all models, and thus in the (pre-)training data, highlights the need for regular updates to maintain the currency and accuracy of the (pre-)training data and the models.

RQ3. Can knowledge editing methods improve the accuracy and consistency of LLMs regarding real-world time-sensitive facts? Despite satisfactory performance on synthetic target datasets in the literature, knowledge editing methods show limitations in updating LLMs regarding real-world knowledge or improving their consistency. The model-dependent performance of the methods highlights the importance of selecting the appropriate editing technique based on the specific model in use. Furthermore, editing the knowledge in a repository requires three types of operations (Dignum and van de Riet, 1992): a) *updating* an existing value attribute with a new value; b) *deleting* a property/attribute thoroughly; and c) *adding* a completely new property/attribute. However, studies on editing the knowledge in LLMs (Yao et al., 2023; Mazzia et al., 2023; Zhang et al., 2023) mostly focus on *updating* operation of an existing knowledge only. This underscores the necessity for tailored approaches when editing LLMs with new knowledge to ensure accuracy and reliability.

6 Conclusion

We have investigated the process of keeping LLMs' knowledge up-to-date and presented an approach to dynamically benchmarking this knowledge via Wikidata. Dynamic benchmarks are a promising solution to address the known limitations of static benchmarks, such as outdatedness and data contamination.

Our results indicate that LLMs differ from traditional knowledge repositories, making it important to investigate what types of knowledge these models can reliably manage and what types of querying and alignment operations they support. We encourage further community engagement to expand DyKnow into a current and active benchmark.

Acknowledgement

We acknowledge the support of the MUR PNRR project FAIR - Future AI Research (PE00000013) funded by the NextGenerationEU.

Limitations

The benchmark is designed based on the Wikidata knowledge base. Other sources can be included to enrich the diversity of the domains and facts in the

benchmark. The performance of the editing methods on the ripple effect of edited time-sensitive facts is not evaluated in this work. The evaluated editing methods are focused on updating the LLMs and do not consider the other operations of removing the knowledge from the model or adding knowledge to the LLM. Lastly, the evaluations of editing methods are limited due to a lack of computation resources, as we could not experiment with larger open-source models.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. 2023. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*.

Davis Brown, Charles Godfrey, Cody Nizinski, Jonathan Tu, and Henry Kvinge. 2023. Robustness of edited neural networks. *arXiv preprint arXiv:2303.00046*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2023a. Evaluating the ripple effects of knowledge editing in language models. *arXiv preprint arXiv:2307.12976*.

Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. 2023b. Crawling the internal knowledge-base of language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1856–1869, Dubrovnik, Croatia. Association for Computational Linguistics.

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

F. Dignum and R.P. van de Riet. 1992. Addition and removal of information for a knowledge base with incomplete information. *Data & Knowledge Engineering*, 8(4):293–307.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.

Vivek Gupta, Pranshu Kandoi, Mahek Vora, Shuo Zhang, Yujie He, Ridho Reinanda, and Vivek Srikumar. 2023. TempTabQA: Temporal question answering for semi-structured tables. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2431–2453, Singapore. Association for Computational Linguistics.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. 2023. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. *arXiv preprint arXiv:2301.04213*.

Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip S. Yu, and Zhijiang Guo. 2024. Towards understanding factual knowledge of large language models. In *The Twelfth International Conference on Learning Representations*.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. 2022. 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*, pages 6237–6250, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A Smith, Yejin Choi, and Kentaro Inui. 2022. Realtime qa: What's the answer right now? *arXiv preprint arXiv:2207.13332*.

Zhoubo Li, Ningyu Zhang, Yunzhi Yao, Mengru Wang, Xi Chen, and Huajun Chen. 2024. Unveiling the pitfalls of knowledge editing for large language models. In *The Twelfth International Conference on Learning Representations*.

Qing Lyu, Kumar Shridhar, Chaitanya Malaviya, Li Zhang, Yanai Elazar, Niket Tandon, Marianna Apidianaki, Mrinmaya Sachan, and Chris Callison-Burch. 2024. Calibrating large language models with sample consistency. *arXiv preprint arXiv:2402.13904*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Vittorio Mazzia, Alessandro Pedrani, Andrea Caciolai, Kay Rottmann, and Davide Bernardi. 2023. A survey on knowledge editing of neural networks. *arXiv preprint arXiv:2310.19704*.

Sachin Mehta, Mohammad Hossein Sekhavat, Qingqing Cao, Maxwell Horton, Yanzi Jin, Chenfan Sun, Iman Mirzadeh, Mahyar Najibi, Dmitry Belenko, Peter Zatloukal, et al. 2024. Openelm: An efficient language model family with open-source training and inference framework. *arXiv preprint arXiv:2404.14619*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2021. Fast model editing at scale. In *International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.

Yuval Pinter and Michael Elhadad. 2023. Emptying the ocean with a spoon: Should we edit models? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15164–15172, Singapore. Association for Computational Linguistics.

Gwenyth Portillo Wightman, Alexandra Delucia, and Mark Dredze. 2023. Strength in numbers: Estimating confidence of large language models by prompt agreement. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 326–362, Toronto, Canada. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Roger Ratcliff. 1990. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285.

Ali Shirali, Rediet Abebe, and Moritz Hardt. 2023. A theory of dynamic benchmarks. In *The Eleventh International Conference on Learning Representations*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2023. Head-to-tail: How knowledgeable are large language models (llm)? aka will llms replace knowledge graphs? *arXiv preprint arXiv:2308.10168*.

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

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Peng Wang, Ningyu Zhang, Xin Xie, Yunzhi Yao, Bozhong Tian, Mengru Wang, Zekun Xi, Siyuan Cheng, Kangwei Liu, Guozhou Zheng, et al. 2023. Easyedit: An easy-to-use knowledge editing framework for large language models. *arXiv preprint arXiv:2308.07269*.

Yifan Wei, Yisong Su, Huanhuan Ma, Xiaoyan Yu, Fangyu Lei, Yuanzhe Zhang, Jun Zhao, and Kang Liu. 2023. MenatQA: A new dataset for testing the temporal comprehension and reasoning abilities of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1434–1447, Singapore. Association for Computational Linguistics.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. *arXiv preprint arXiv:2305.13172*.

Xunjian Yin, Baizhou Huang, and Xiaojun Wan. 2023. ALCUNA: Large language models meet new knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1397–1414, Singapore. Association for Computational Linguistics.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, et al. 2023. Kola: Carefully benchmarking world knowledge of large language models. *arXiv preprint arXiv:2306.09296*.

Zihan Zhang, Meng Fang, Ling Chen, Mohammad-Reza Namazi-Rad, and Jun Wang. 2023. How do large language models capture the ever-changing world knowledge? a review of recent advances. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8289–8311, Singapore. Association for Computational Linguistics.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? *ArXiv*, abs/2305.12740.

A Appendix

Subject Category		Subject Entity	Property	Prompt Template with [Place Holder]
Countries		Italy, United States of America, China, Germany, Japan, India, United Kingdom, France, Brazil, Canada, Russia, Mexico, South Korea, Australia, Spain, Indonesia, Turkey, Netherlands, Saudi Arabia, Poland, Belgium, Argentina, Sweden, Ireland, Norway, Austria, Israel, Thailand, United Arab Emirates, Singapore, Bangladesh, Philippines, Vietnam, Malaysia, Denmark, Egypt, Nigeria, South Africa, Iran, Colombia, Romania, Chile, Pakistan, Czech Republic, Finland, Iraq, Portugal	***(Head of State)***	❑ *The [Title] of [Country] is* ❑ *The name of the current [Title] of the [Country] is* ❑ *The position of the [Title] of [Country] is currently held by*
			President	
			King	
			Monarch	
			Emperor	
			Supreme Leader	
			(Head of Gov.)	
			Prime Minister	
			Premier of Republic	
			Federal Chancellor	
Athletes	Soccer Player	Cristiano Ronaldo, Lionel Messi Neymar Jr., Kylian Mbappé, Karim Benzema, Erling Haaland, Mohamed Salah, Sadio Mané, Kevin De Bruyne,Harry Kane	Sports Team	❑ *[Athlete] is currently playing for* ❑ *The football team [Athlete] currently plays for is* ❑ *The current football club of [Athlete] is*
	Basketball Player (NBA)	Stephen Curry, Kevin Durant, LeBron James, Nikola Jokic, Bradley Beal, Giannis Antetokounmpo, Damian Lillard, Kawhi Leonard, Paul George		❑ *[Athlete] is currently playing for* ❑ *The NBA team [Athlete] currently plays for is* ❑ *The basketball team [Athlete] currently plays for is*
	F1 Driver	Max Verstappen, Lewis Hamilton, Fernando Alonso, Sergio Pérez, Charles Leclerc, Lando Norris, Carlos Sainz Jr., George Russell, Pierre Gasly		❑ *[Athlete] is currently racing for* ❑ *The Formula 1 team [Athlete] currently drives for is* ❑ *The team [Athlete] is currently racing for in Formula 1 is*
Private / Public Organi-zations	Company	Walmart, Saudi Aramco, Amazon, ExxonMobil, Apple, Shell, CVS Health, Volkswagen Group, Alphabet Inc., Toyota, TotalEnergies, Glencore, BP, Cencora, Inc., Microsoft, Gazprom, Mitsubishi, Ford Motor Company	CEO	❑ *The position of Chief Executive Officer at [Company] is currently held by* ❑ *The CEO position at [Company] is currently held by* ❑ *The current CEO of [Company] is*
	Organi-zation	United Nations, World Bank, International Atomic Energy Agency, North Atlantic Treaty Organization, International Olympic Committee	Director / Manager	❑ *The name of the current director at [Organization] is* ❑ *The director position at [Organization] is currently held by* ❑ *The position of director at [Organization] is currently held by*
			Headquarters Location	❑ *The main office location of [Organization] is* ❑ *The head office of [Organization] is located in* ❑ *The central location of [Organization] is*
			Chairperson	❑ *The name of the current chairperson at [Organization] is* ❑ *The position of chairperson at [Organization] is currently held by* ❑ *The chairperson position at [Organization] is currently held by*
			General secretary	❑ *The position of general secretary at [Organization] is currently held by* ❑ *The name of the current general secretary at [Organization] is* ❑ *The general secretary position at [Organization] is currently held by*

Table 3: The list of subject entities and properties as time-sensitive facts used in benchmarking the LLMs. We used three prompt templates for each category.

(Year) model	Correct	Outdated	Irrelevant
(2019) GPT-2	15%	24%	61%
(2020) GPT-3	22%	32%	46%
(2020) T5	5%	12%	83%
(2021) GPT-J	29%	35%	36%
(2022) Bloom	24%	36%	40%
(2022) Flan-T5	13%	32%	55%
(2023) Llama-2	35%	32%	33%
(2023) Falcon	31%	35%	34%
(2023) Mistral	38%	33%	29%
(2023) Mixtral	36%	33%	31%
(2024) OLMo 1B	23%	27%	50%
(2024) OLMo 7B	35%	29%	36%
(2024) Llama-3	37%	34%	29%
(2024) OpenELM 270M	5%	13%	82%
(2024) OpenELM 1.1B	24%	33%	43%
(2024) OpenELM 3B	31%	31%	38%
(2022) ChatGPT	56%	35%	9%
(2023) GPT-4	77%	15%	8%
(2023) Llama-$2_{C.}$	47%	35%	18%
(2023) Falcon$_{I.}$	38%	40%	22%
(2023) Vicuna	44%	32%	24%
(2023) Mistral$_{I.}$	51%	29%	20%
(2023) Mixtral$_{I.}$	59%	31%	10%
(2024) Llama-$3_{I.}$	69%	17%	14%

Table 4: Benchmarking 24 LLMs with time-sensitive knowledge. Differently from Table 1, the scores are computed by averaging the model performance across the three prompts. The table presents the percentage of **C**orrect answers that are valid and up-to-date, **O**utdated answers that are not valid anymore, and **I**rrelevant outputs. Models below the dashed line were prompted with an additional prefix "Answer with the name only".

(Year) Model	Prompt Agreement (%)
(2019) GPT-2	11%
(2020) GPT-3	9%
(2020) T5	19%
(2021) GPT-J	25%
(2022) Bloom	25%
(2022) Flan-T5	49%
(2023) Llama-2	24%
(2023) Falcon	31%
(2023) Mistral	34%
(2023) Mixtral	29%
(2024) OLMo 1B	20%
(2024) OLMo 7B	23%
(2024) Llama-3	25%
(2024) OpenELM 270M	4%
(2024) OpenELM 1.1B	22%
(2024) OpenELM 3B	27%
(2022) ChatGPT	98%
(2023) GPT-4	94%
(2023) Llama-2$_{C.}$	82%
(2023) Falcon$_{I.}$	66%
(2023) Vicuna	69%
(2023) Mistral$_{I.}$	87%
(2023) Mixtral$_{I.}$	88%
(2024) Llama-3$_{I.}$	84%

Table 5: The level of prompt agreement for each model across three prompts for each time-sensitive question. The agreement is computed as the percentage of times a model gives the same answer to all three prompts. Subscripts $I.$ and $C.$ stand for *Instruct* and *Chat*, respectively.

Model	#Outdated Facts	Knowledge Editing			
		Modifying Parameters		Preserving Parameters	
		ROME	MEMIT	SERAC	IKE
(2019) GPT-2	54	12%	25%	3%	39%
(2021) GPT-J	60	10%	71%	0%	95%‡
(2023) Llama-2$_{C.}$	48	3%	72%	27%	17%
(2023) Mistral$_{I.}$	41	0%	0%	—	86%‡

Table 6: Performance of different methods for aligning the outdated knowledge in 4 LLMs, by **paraphrase success** (Meng et al., 2022a,b). ‡ indicates successful alignment of more than 85% outdated knowledge.

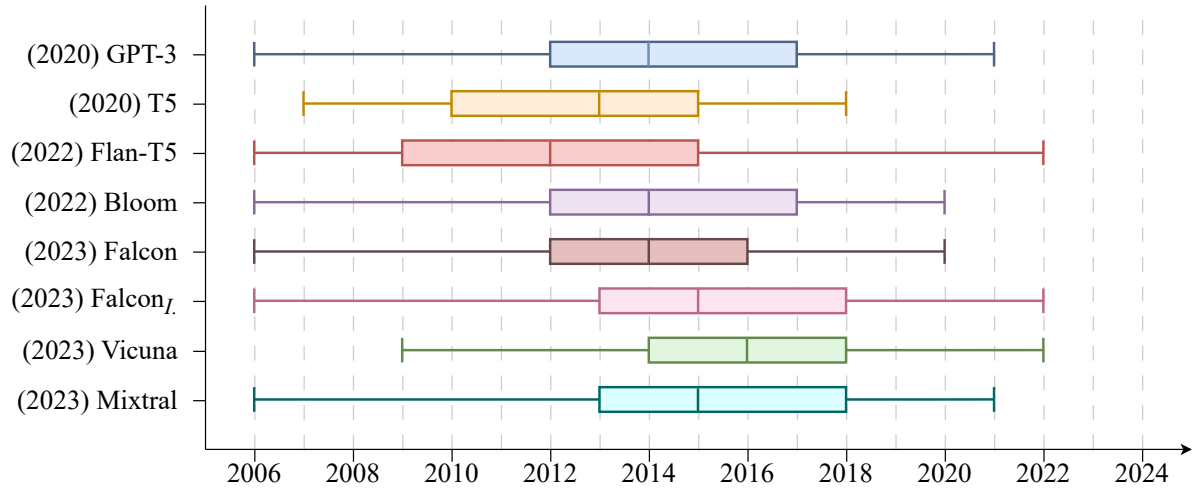

Figure 5: Approximating the temporal period of the data used for (pre-)training the models according to their correct and outdated outputs to our time-sensitive factual questions. The y-axis presents the evaluated LLMs with their release year in parentheses. The box plots present the distribution of the generated responses for each LLM according to their validity interval. Each box plot shows the interquartile range of the responses, with whiskers extending to the minimum and maximum dates.

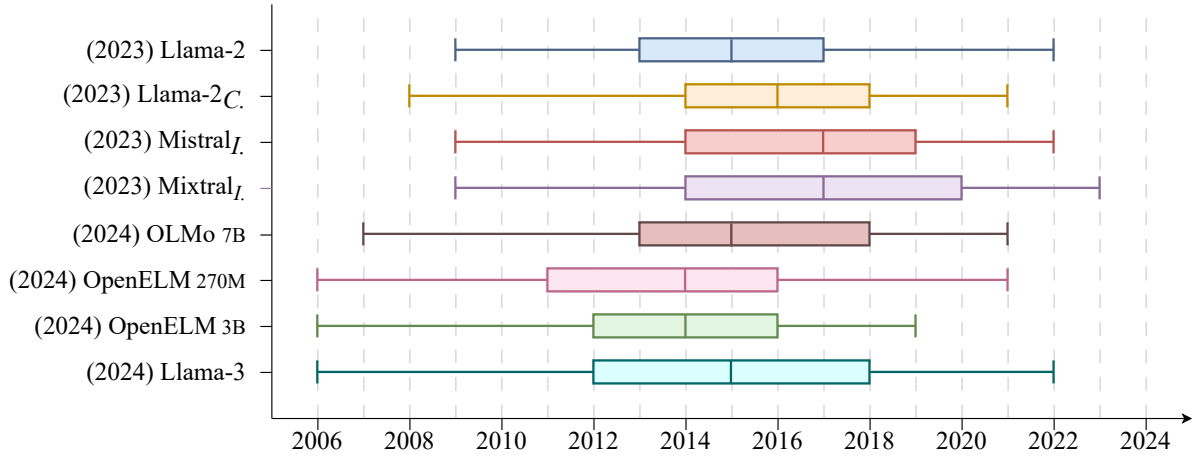

Figure 6: Approximating the temporal period of the data used for (pre-)training the models according to their correct and outdated outputs to our time-sensitive factual questions. The y-axis presents the evaluated LLMs with their release year in parentheses. The box plots present the distribution of the generated responses for each LLM according to their validity interval. Each box plot shows the interquartile range of the responses, with whiskers extending to the minimum and maximum dates.

A.1 Editing Methods Implementation Details

We present more details regarding the editing methods we experiment with:

- **ROME** (Meng et al., 2022a) locates the corresponding parameters for each factual knowledge in the feed-forward layers of the model via causal mediation analysis (Geva et al., 2021). It then inserts a new key-value association (representing the edited knowledge) in the original parameters by formulating it as a least squares problem with a linear equality constraint.

- **MEMIT** (Meng et al., 2022b) expands ROME for applying multiple edits at once. While ROME can perform one edit by operating on one layer at a time, MEMIT applies several edits by operating on several layers in one intervention.

- **SERAC** (Mitchell et al., 2022) uses an external memory to store the new facts, and a classifier to measure the similarity of the question prompt with the stored facts in the memory. If there is no match between the question prompt and the facts in the memory, the primary LLM is selected to generate the final output. In cases of a match between the question prompt and a new fact in the memory, a secondary model (a smaller language model) generates the response grounded on the matching new fact.

- **IKE** (Zheng et al., 2023) is based on in-context learning. To answer the question q^* by the new (up-to-date) attribute value y^*, this method constructs a prompt consisting of the question, the corresponding up-to-date fact f^*, and a context segment (q^*, f^*, C). The context consists of k triplets ($C = \{c_1,...,c_k\}$) of facts, corresponding questions, and values $c_i = (f_i, q_i, y_i)$. The triplets are retrieved based on the cosine similarity with (q^*, f^*, y^*) from a pre-defined pool and serve as examples for using the information in f_i to answer q_i. The prompt (q^*, f^*, C) is then presented to the model to output an answer. Note that this technique does not represent a realistic scenario, since it requires the relevant and up-to-date fact f^* for each question to be deterministically provided to the model.

The experiments were applied to Huggingface gpt2-xl, EleutherAI/gpt-j-6b, meta-llama/Llama-2-7b-chat-hf, and mistralai/Mistral-7B-Instruct-v0.1. Regarding the knowledge editing methods, we followed EasyEdit framework (Wang et al., 2023) for ROME, MEMIT, SERAC, and IKE and utilized the default model-specific configuration of the hyper-parameters. For Llama-$2_{C.}$ and Mistral$_{I.}$, we considered the configurations for the non-chat and non-instruct versions, respectively. Training SERAC for GPT-2, GPT-J, and Llama-$2_{C.}$ required one NVIDIA A100 with 80 GiB. Model inference was performed on two NVIDIA GeForce RTX 3090 with 24.5 GiB each, except for Mixtral which required three NVIDIA A100 with 80 GiB. Regarding ROME, the edits are applied sequentially and the original weights of the models are not reverted after each edit. That is, as a knowledge repository, the models must be able to retain all the edits. Regarding MEMIT, all the changes are applied in one intervention. Regarding SERAC, the memory consists of the new facts to be edited for each model.