Carpe Diem[*] 🎁: On the Evaluation of World Knowledge in Lifelong Language Models

Yujin Kim[1]   **Jaehong Yoon**[2]   **Seonghyeon Ye**[1]   **Sangmin Bae**[1]   **Namgyu Ho**[1]
Sung Ju Hwang[1][†]   **Se-young Yun**[1][†]
[1]KAIST AI   [2]UNC Chapel Hill
{yujin399, sjhwang82, yunseyoung}@kaist.ac.kr

Abstract

The dynamic nature of knowledge in an ever-changing world presents challenges for language models trained on static data; the model in the real world often requires not only acquiring new knowledge but also overwriting outdated information into updated ones. To study the ability of language models for these time-dependent dynamics in human language, we introduce a novel task, *EvolvingQA*, a temporally evolving question-answering benchmark designed for training and evaluating LMs on an evolving Wikipedia database. The construction of *EvolvingQA* is automated with our pipeline using large language models. We uncover that existing continual learning baselines suffer from updating and removing outdated knowledge. Our analysis suggests that models fail to rectify knowledge due to small weight gradients. In addition, we elucidate that language models particularly struggle to reflect the change of numerical or temporal information. Our work aims to model the dynamic nature of real-world information, suggesting faithful evaluations of the evolution-adaptability of language models. Our data construction code and dataset files are available at https://github.com/kimyuji/EvolvingQA_benchmark.

1 Introduction

Large language models (LLMs) (Radford et al., 2018; Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023) have demonstrated remarkable capabilities in encoding vast amounts of knowledge in massive training data, which can be applied for downstream tasks such as knowledge-intensive question-answering and multi-hop reasoning. However, knowledge is not static: scientific discoveries, cultural trends, and linguistic creativity are constantly updated and edited as the world
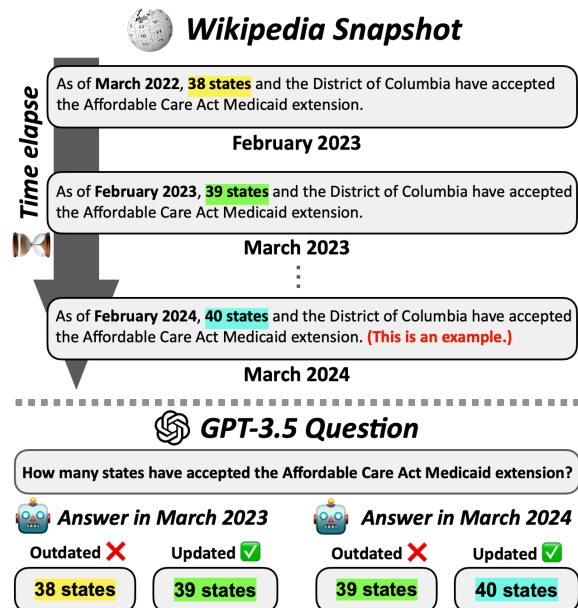

Figure 1: An overview of our evaluation benchmark, EvolvingQA. Our benchmark employs LLM to generate question-answer pairs based on the changes in Wikipedia's snapshots, effectively capturing the temporal evolution of the knowledge base.

changes. Current LLMs are trained on static data, implying that the encoded knowledge could go wrong as time passes, which affects their reasoning abilities (Dhingra et al., 2022).

Meanwhile, previous research has shown that language models pre-trained on reliable knowledge sources such as Wikipedia can substitute knowledge bases by storing knowledge in their parameters and be applied to various downstream tasks (Petroni et al., 2019; Roberts et al., 2020). To keep these models up to date with evolving world knowledge, it is desirable to apply continuous pre-training rather than periodically re-training from scratch.

Continued learning of existing models over sequential time-varying data remains one of the critical challenges in machine learning and has been

---

[*]*Carpe diem* is a Latin phrase that translates to "Live in the present" in English. It encourages individuals to make the most of the present🎁 moment.
[†]Corresponding authors

Attribute	EvolvingQA (Ours)	CKL (Jang et al., 2021)	TemporalWiki (Jang et al., 2022)	StreamingQA (Liška et al., 2022)	RealTimeQA (Kasai et al., 2023)
Edited knowledge	✓	✓	✗	✗	✗
Automatic construction	✓	✗	✓	✗	✗
# of timestamps	6 (Unlimited)	2	4 (Unlimited)	4	(Unlimited)
Available Tasks	QA	Slot-filling	Slot-filling	QA	QA

Table 1: Comparison of our benchmark and existing benchmarks for temporal alignment. Detailed descriptions for each attribute are presented in Appendix B.

widely discussed in previous literature, often referred to as continual learning (CL) (Thrun, 1995; Li and Hoiem, 2016a; Lee et al., 2017a; Wang et al., 2022) or lifelong learning. This learning paradigm addresses the problem of learning on multiple tasks/data sequentially, assuming that the data from the previous session is inaccessible when starting the next training session. The primary goal is to preserve previously acquired knowledge while learning new concepts.

However, in real-world scenarios, consistent accumulation of world knowledge while *forgetting outdated knowledge is desirable* due to changes in world knowledge. Models are required not only to learn *new* information but also to forget or *update* outdated information[1]. For example, the knowledge from 2017 that "Donald Trump is the president of the US." became outdated in 2021, when it was substituted by the updated knowledge "Joe Biden is the president of the US." Several benchmarks have been introduced to evaluate language models on temporally changing knowledge (Jang et al., 2021, 2022; Neelam et al., 2022; Liška et al., 2022; Kasai et al., 2023). However, these fall short in providing holistic evaluations of knowledge preservation *and* modification in the context of real-world applications. While Jang et al. (2021, 2022) address both changed and unchanged knowledge, models are evaluated using template-based knowledge probing (i.e., LAMA task (Petroni et al., 2019)), which may not represent applicability in the real world. Kasai et al. (2023) focus on evaluating new and updated knowledge, thereby failing to assess catastrophic forgetting of previous knowledge after updated knowledge acquisition. We provide a comprehensive comparison of benchmarks in Table 1.

Our goal is to create a benchmark for holistic evaluation of the temporal adaptation capabilities of language models. We propose **EvolvingQA**, a novel benchmark for pre-training and evaluating

LMs over evolving Wikipedia data. We propose an automated pipeline to construct our benchmark using LLMs, which allows us to extend our benchmark to many time steps and to easily update the benchmark into the future, as depicted in Figure 1. We use the question-answering (QA) task for downstream evaluation to measure continual learning that translates into real-world applicability. We find that continual pre-training baselines (1) suffer from catastrophic forgetting and (2) fail to forget outdated knowledge (3) or incorporate updated knowledge, highlighting the relevance of our benchmark. We provide comprehensive analyses on why and how such circumstances occur.

Our contributions are as follows:

- We propose a new benchmark to evaluate LMs on preserving time-invariant knowledge while integrating changes through continual pre-training. Our benchmark incorporates open-domain question-answering, which is an intuitive and practical downstream task. Our dataset construction pipeline is automated by using LLMs, which can be generated at low cost.

- Our experimental results on EvolvingQA show that the baselines struggle to learn updated knowledge and forget previously learned outdated knowledge.

- We provide in-depth analyses on why and how the existing baselines fail to predict updated information. The language models especially struggle to update numerical or temporal knowledge, because the models' gradient is not significant enough to forget outdated knowledge when learning updated knowledge.

2   EvolvingQA

In this section, we introduce EvolvingQA, a novel benchmark for evaluating LM's ability to forget and update dynamically evolving knowledge. EvolvingQA is divided into continual pre-training cor-

---

[1] *New* knowledge refers to added knowledge which was previously nonexistent, while *updated* knowledge refers to added knowledge which invalidates previous knowledge.
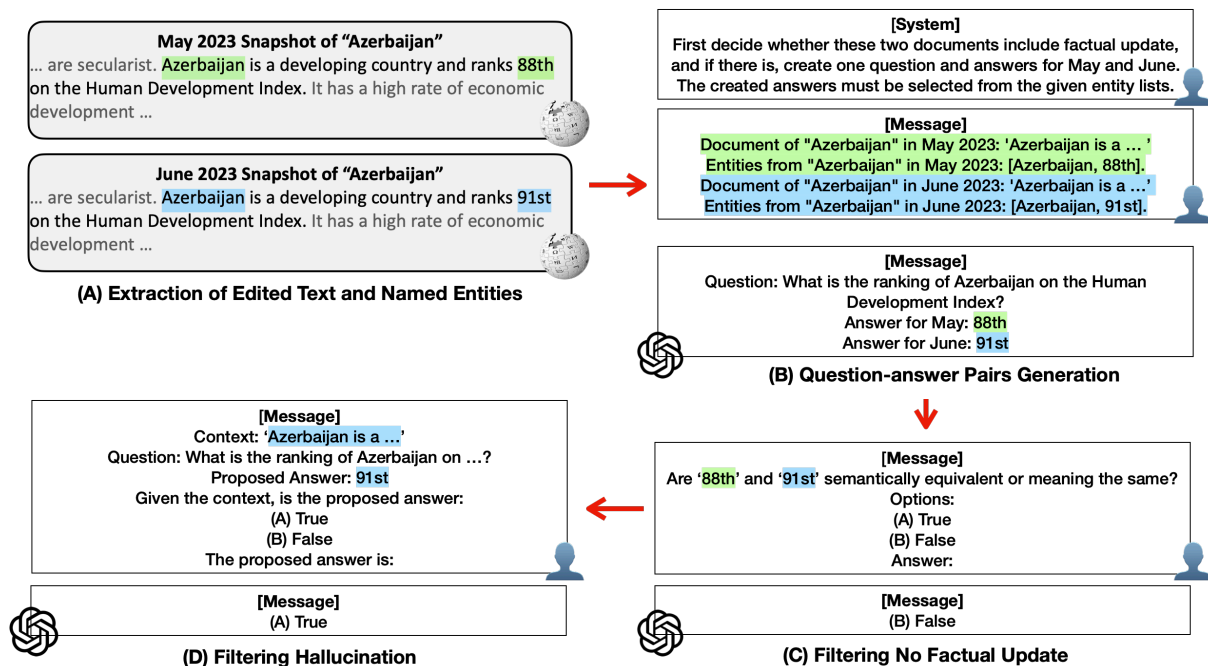
Figure 2: Construction pipeline of EDITED. The final question-answers pair after filtering processes in this Figure is included in EDITED06. The full description of the pipeline is in Section 2.2 and Appendix C.

pora and evaluation dataset. For continual pre-training corpora, we collect consecutive Wikipedia snapshots and conduct heuristic filtering. For evaluation dataset, we collect a QA dataset through automatic generation and validation using LLM. Since both pre-training and evaluation data can be collected automatically, EvolvingQA could be extended to future time steps.

2.1 Continual Pre-training Dataset

We collect CHANGED sets, pre-training corpora consisting of changes between two consecutive Wikipedia snapshots. We exclude Wikipedia articles with minimal updates from CHANGED sets, focusing more on knowledge that have undergone sufficient changes. Specifically, we only select Wikipedia articles that the updated part is more than the length of 500 characters as our continual pre-training dataset. We call these resulting subsets a CHANGED set. For example, the CHANGED03 set includes parts of Wikipedia articles of March 2023 that were modified from February 2023. The number of topics in different corpus from each time step is shown in Table 4. We process CHANGED to follow T5 pre-training objective. Particularly, following Roberts et al. (2020), we use salient span masking which set preparing the input as a text

in which named entities and dates [2] are masked, and the output is then set as the corresponding unmasked entities and dates. A sample of input and output from CHANGED03 is reported in Figure 8.

2.2 EvolvingQA benchmark

We construct a question-answering benchmark to measure the model's capability of answering correctly while learning temporally changing knowledge. To measure how the language models 1) prevent catastrophic forgetting of old knowledge, 2) acquire new knowledge, and 3) edit their outdated knowledge into updated knowledge, we construct UNCHANGED, NEW, and EDITED evaluation sets, respectively.

We extract parts of Wikipedia articles that are unmodified, new, and edited, using the `difflib` library. We then prompt GPT-3.5[3] to generate question-answer pairs using the extracted parts. GPT-3.5 is conditioned to select answers from the given named entities, to ensure short-form answers. Note that the named entities that we provide GPT-3.5 are the ones that our language model is learned to reconstruct during pre-training CHANGED sets. The generated question-answer pairs are provided to GPT-3.5 as input for further filtering.

---

[2] We use `en_core_web_trf` model to extract named entities and dates provided from spaCy (https://spacy.io/).
[3] We use GPT-3.5-turbo-0613 provided by OpenAI API.

Dataset	03	04	05	06	07	08
UNCHANGED	49,504	49,504	49,504	49,504	49,504	49,504
NEW	29,680	32,954	31,487	32,845	38,584	32,559
EDITED	7,293	2,259	1,889	1,708	1,672	8,462

Table 2: The number of question-answer pairs for evaluation.

UNCHANGED The UNCHANGED evaluation set aims to measure how well the models maintain the knowledge obtained initially, even after learning the series of upcoming knowledge. We gather Wikipedia articles from the February 2023 snapshot that have not altered during the next six months. We then utilize the unaltered parts to prompt GPT-3.5 as context to create question-answer pairs. We condition GPT-3.5 to select the ground truth answer to be one of the given entities that were masked for pre-training input. The resulting UNCHANGED set is used to evaluate models on all time steps.

In order to make language models answer a given question in a desired format, fine-tuning pre-trained models on question-answering task is required. We extract 80K additional question-answer pairs from unchanged topics to construct a fine-tuning dataset, and we ensure it is disjoint with UNCHANGED set. Consequently, continually pre-trained models are fine-tuned using the 80K unchanged pairs, and then evaluated with UNCHANGED, NEW, and EDITED of the corresponding time step. The resulting statistic of our benchmark is reported in Table 2.

NEW The NEW evaluation set shows how well the language models learn new knowledge that does not contradict the previously learned knowledge. We use CHANGED set of corresponding time steps to construct NEW evaluation set. For example, to evaluate a model continually pre-trained until May 2023 (i.e., a model continually pre-trained from initial time step to CHANGED05), we use the NEW 05 set which consists of question-answer pairs extracted from CHANGED05. Similar to UNCHANGED, we prompt GPT-3.5 to create question-answer pairs, while conditioning answers should be selected from the given entities.

EDITED The EDITED evaluation set measures how the models forget outdated knowledge and learn updated knowledge when the previously learned knowledge gets outdated by the articles edited. The overview of our EDITED construction pipeline is depicted in Figure 2. In order to create question-answer pairs that reflect the edit of knowl-

---

[System]
You are a helpful assistant and will be provided with two documents that are parts of Wikipedia articles of the same topic but written in February 2023 and March 2023. First, decide whether these two documents include any factual update. If there is no factual update, simply write "no factual update" and do not write anything else. If there is any factual update between the two, then create ONE short question and TWO answers that the answer for February and the answer for March are different. The answer for the created pair MUST be selected from one of the entities from the given list.

[User]
Document of "Alaska" in February 2023: 'If it was an independent nation would be the 16th largest country in the world, larger than Iran.'
Entities from "Alaska" in February 2023: [16th, Iran].
Document of "Alaska" in March 2023: 'If it was an independent nation would be the 17th largest country in the world, larger than Iran.'
Entities from "Alaska" in March 2023: [17th, Iran].

[Assistant] Question: What is the ranking of Alaska if it was an independent nation?
Answer1: 16th
Answer2: 17th

[User]
Document of "Azerbaijan" in February 2023: 'Azerbaijan is a developing country and ranks 88th on the Human Development Index.'
Entities from "Azerbaijan" in February 2023: [Azerbaijan, 88th].
Document of "Azerbaijan" in March 2023: 'Azerbaijan is a developing country and ranks 91st on the Human Development Index.'
Entities from "Azerbaijan" in March 2023: [Azerbaijan, 91st].

Figure 3: An example of the prompt we use in generating QA pairs in EDITED set. The blue-colored messages are one-shot demonstration to make sure GPT-3.5 follow the instruction more accurately and generate question-answer instances in a desired format.

edge, we collect the revised parts of Wikipedia articles, and provide GPT-3.5 the original part (i.e., outdated part as of current time step) and the corresponding revised part (i.e., updated part as of current time step). The prompt we used to generate the QA pairs is described in Figure 3. To filter out cases where the update only includes stylistic change or grammatical correction, we use system command to condition GPT-3.5 to determine if the context from two consecutive time steps does include factual updates. We also condition the answers should be one of the provided candidate entities for short and precise answers. Lastly, we provide GPT-3.5 one-shot example of question-answer generation for better alignment. The resulting EDITED QA instance generated by GPT-3.5 includes a question,

an OUTDATED answer, and an UPDATED answer.

After extracting question-answer pairs, we go through further filtering process to remove the hallucination and bias of GPT-3.5 by asking whether the answer is correct given the context and question, following Kadavath et al. (2022). The details of prompts and filtering methods used in EvolvingQA construction pipeline are described in the Appendix C.

3 Experiment

3.1 Training Details

We utilize 737M T5-large (Raffel et al., 2020), specifically `google/t5-large-ssm` pre-trained checkpoint from Roberts et al. (2020). In EvolvingQA continual learning framework, we begin with an initial checkpoint (INITIAL), which is further pre-trained on the entire Wikipedia snapshot of February 2023. We then continual pre-train sequentially on CHANGED sets, using the learning rate of 1e-3 and gradient accumulation by 3 with a batch size of 5. To evaluate model's knowledge on each time step, we fine-tune continually pre-trained models on the QA train set composed of unchanged knowledge. We use 1e-5 for the learning rate with a batch size of 32 and train for a single epoch to avoid memorization. Then, we evaluate it on UNCHANGED, NEW, and EDITED evaluation sets of each corresponding time step. During inference, greedy decoding is used, and we pre-process the decoded output and ground truth answer by changing it into lowercase and removing punctuation. This process is applied identically across all time steps.

3.2 Baselines

INITIAL INITIAL is a starting checkpoint, before any continual pre-training on the following CHANGED sets. We pre-train T5-large using the entire Wikipedia snapshot of February 2023. The checkpoint of INITIAL serves as the initial checkpoint of all the other CL methods.

FULL We start from INITIAL and continue pre-training on CHANGED sets sequentially. The full model is updated without freezing any parameter. This approach is similar to domain-adaptive pre-training proposed from Gururangan et al. (2020).

K-Adapter K-Adapter (Wang et al., 2020) is an architecture-based continual learning method, which trains additional adapters to the LM while freezing the original parameters. We use $k=2$ where the adapters are inserted after the second and the last layers. We only freeze the encoder part of encoder-decoder network and update decoder and adapters following Jang et al. (2021).

LoRA We implement parameter-efficient training method, LoRA (Hu et al., 2021), which trains rank decomposition matrices of each layer while freezing the original parameter. We use $r=4$ and adapt W_q and W_v in self-attention layer. We only freeze the encoder part of encoder-decoder network and update decoder and LoRA module.

DPR We compare baselines with the retrieval-based method proposed by Karpukhin et al. (2020), which encodes passages into dense representations and retrieves context representations closest to the question representations. Namely, the most relevant context for each question is determined by calculating the dot product of the question embedding with all the context embeddings from the knowledge base. The retrieved contexts are used as context in open-book question answering. We fine-tune the pre-trained T5-large using QA pairs of unchanged knowledge (i.e., UNCHANGED pairs for fine-tuning) providing the context, to create the reader model. We use `facebook/dpr-ctx_encoder-single-nq-base` and `facebook/dpr-question_encoder-single-nq-base` models to create context and question embeddings, respectively.

3.3 Results

Table 3 reports the result of baselines through sequentially learning Wikipedia articles from CHANGED03 to CHANGED08 starting from INITIAL. We measure Exact Match (EM) and F1 score, and F1 score is calculated by counting the common tokens between predicted answer and ground truth answer. We additionally provide visualization of the result of F1 scores in Figure 4. The result shows that all the CL baselines struggle with catastrophic forgetting, while FULL forgets the unchanged knowledge the most. FULL also struggle from acquiring NEW knowledge compared to other methods, and we conjecture that if the knowledge from different time steps is not learned with isolated parameters, it can result in blurring knowledge from different time steps. Meanwhile, K-Adapter and LoRA exhibit comparably high stability and plasticity, since they freeze the encoder and train with additional parameters.

Method	Dataset	EM						F1					
		03	04	05	06	07	08	03	04	05	06	07	08
INITIAL	UNCHANGED	5.17	5.17	5.17	5.17	5.17	5.17	10.37	10.37	10.37	10.37	10.37	10.37
	NEW	4.82	4.97	4.41	5.18	5.23	4.03	8.64	8.82	7.90	8.77	9.02	8.05
	OUTDATED ↓	2.30	2.19	2.68	2.21	2.80	2.65	7.30	7.15	7.88	6.99	7.71	7.58
	UPDATED ↑	2.41	2.27	2.59	2.57	2.34	2.33	7.35	6.91	7.48	7.28	6.71	7.28
FULL	UNCHANGED	3.78	3.62	3.37	3.33	3.28	3.11	8.41	8.20	7.95	7.86	7.79	7.66
	NEW	5.23	4.64	4.27	4.78	4.68	3.43	9.45	8.69	8.22	8.56	8.44	7.53
	OUTDATED ↓	2.43	2.15	2.82	1.96	2.70	2.10	7.22	7.06	8.09	6.62	7.37	7.03
	UPDATED ↑	2.23	2.49	2.33	2.47	2.19	2.05	7.73	7.78	8.04	7.59	7.36	7.59
K-Adapter (Wang et al., 2020)	UNCHANGED	4.64	4.55	4.44	4.40	4.43	4.45	9.47	9.44	9.40	9.37	9.35	9.40
	NEW	5.52	5.42	4.83	5.29	5.42	4.25	9.83	9.64	8.80	9.41	9.59	8.83
	OUTDATED ↓	2.44	2.68	2.64	2.42	2.80	2.58	7.62	7.78	7.98	7.78	7.72	7.80
	UPDATED ↑	2.43	2.79	2.95	2.97	2.60	2.70	8.02	8.32	8.84	8.36	7.72	8.31
LoRA (Hu et al., 2021)	UNCHANGED	4.65	4.43	4.41	4.39	4.35	4.37	9.45	9.25	9.46	9.27	9.33	9.33
	NEW	5.57	5.32	4.93	5.31	5.46	4.13	9.75	9.51	9.06	9.34	9.71	8.56
	OUTDATED ↓	2.64	2.53	3.04	2.77	2.65	2.55	7.80	7.42	8.40	7.96	7.88	7.87
	UPDATED ↑	2.64	2.87	2.95	2.82	2.70	2.54	8.31	8.16	8.31	8.40	8.11	8.42
DPR (Karpukhin et al., 2020)	UNCHANGED	40.58	40.07	41.62	40.12	39.98	40.00	43.32	42.52	42.95	42.44	41.28	42.52
	NEW	18.54	24.67	22.00	21.33	22.67	23.33	22.91	29.42	25.71	25.18	28.08	27.38
	OUTDATED ↓	4.23	4.01	3.67	4.00	5.33	4.28	10.84	10.73	10.73	10.55	12.56	10.16
	UPDATED ↑	23.87	29.33	19.33	16.67	19.67	21.33	29.74	35.98	21.40	20.60	25.02	25.93

Table 3: The results of question answering task according to baseline methods. Exact match (EM) and F1 score are measured. Note that in ideal setting, the performance of OUTDATED should be as close to zero as possible if the model successfully forgets outdated knowledge. The result is from a single run.
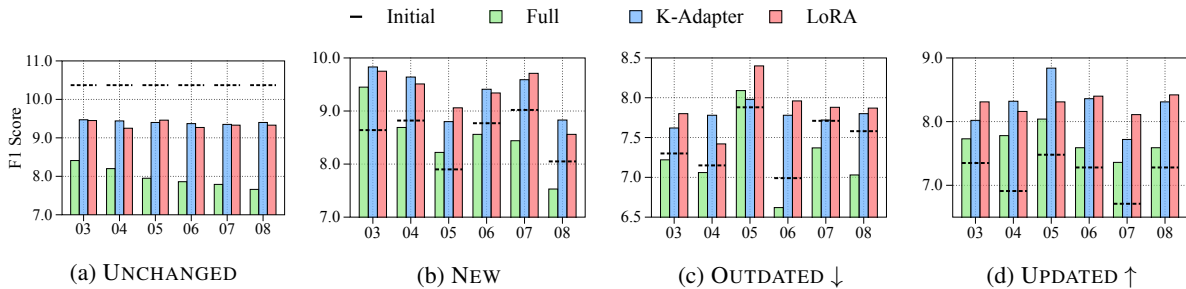

Figure 4: The bar plot that shows the trend of F1 scores through continual learning of CHANGED sets. Note that a single UNCHANGED set is used to evaluate on all time steps.

In contrast, the overall performance in OUTDATED and UPDATED presents that all baselines suffer from forgetting outdated knowledge and acquire updated knowledge. Ideally, the performance for OUTDATED should be close to zero when the model perfectly updated their knowledge. However, most of the baselines result in similar OUTDATED performance with UPDATED performance.

We also conduct additional experiment on shifting QA task into multiple choice answering, where OUTDATED and UPDATED answer are two answer candidates. As shown in Table 5 in Appendix D, the result also indicates that with more than 50% of selecting OUTDATED answer, the models remain outdated.

Meanwhile, DPR shows significant and meaningful result, where performance of OUTDATED is much lower than UPDATED, thus demonstrating our benchmark's accuracy and faithfulness.

3.4 Analysis on EDITED Knowledge

In this section, we delve into a thorough analysis of the reasons and mechanisms behind the failure of language models to update their information through the continual pre-training process.

3.4.1 Gradients of EDITED Knowledge

We conduct an analysis to observe the differing trends in gradient updates when the model processes new or edited information during continual pre-training. Figure 5 illustrates the Frobenius norm of the model's weight gradients when exposed to newly introduced or updated knowledge, specifically using inputs from the CHANGED03 set. This is calculated based on the INITIAL checkpoint,
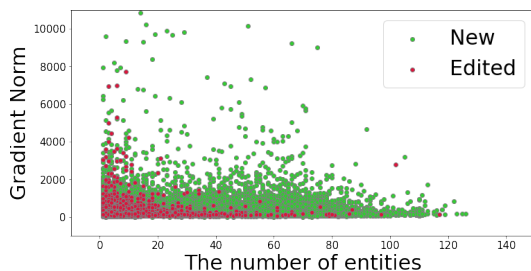
Figure 5: The scatter plot of samples in CHANGED03 according to the number of masked entities and gradient norm. Each dot indicates a sample from either NEW knowledge or EDITED knowledge in CHANGED. The x-axis shows the number of masked entities in a sample. The y-axis shows the Frobenius norm of weight gradients of each sample.

encompassing gradients across all parameters. Similar trends are observable in other time steps as depicted in Appendix F. Notably, when the model is fed with updated knowledge (red color), the norms of the weight gradients are considerably smaller and closer to zero, in contrast to when it processes new knowledge (green color). This suggests that the model's gradient updates are less significant to forget the outdated information when trained with updated information. We hypothesize that this is because the updated information closely resembles the form of the previously learned information, rendering it more recognizable to the model.

3.4.2 Quantitative Analysis of EDITED knowledge

To investigate the types of knowledge that LMs struggle to update, we categorize QA instances into eight distinct types. Figure 6 presents the EM scores for each category, using the FULL method, evaluated on NEW and EDITED sets.

As illustrated in Figure 6 (a), the distribution of EM scores across different categories in the NEW set remains consistent across all time steps. Notably, models that have undergone continual pre-training show enhanced performance in the Culture/Group, Locational, and Art/Media categories. However, as depicted in Figure 6 (b), the models exhibit challenges in accurately predicting knowledge within the numerical and temporal categories for the EDITED set, with EM nearing zero across all time steps. This suggests a notable deficiency in the models' ability to effectively update numerical or temporal information. The descriptions of each category are explained in Appendix G.

4 Discussion

4.1 Knowledge Change in Wikipedia

Wikipedia, a widely-used online encyclopedia, exemplifies collective intelligence with its open editing system. Its monthly snapshots enable tracking of article changes, forming the basis of our dataset. These changes are categorized into three types: (1) updates with recent news or facts, (2) additions or corrections of existing information, and (3) grammatical corrections.

Updates with Recent News involve adding current events or new discoveries, reflecting the evolving nature of world knowledge. It is the most crucial part that our benchmark aims to encompass. Note that such update does not always reflect real-time news immediately.

Additions or Corrections of Existing Information are frequent, involving updates to historical events or figures. While not always reflecting current events, it is important to consider these modification. For the cases where models have learned erroneous or private data, ensuring models to remain accurate and respectful of privacy concerns is significant and challenging. Continuous information revision is key to the development and ethical integrity of language models.

Grammatical Corrections are minimized in our EvolvingQA dataset using heuristic algorithms and LLM validation. However, a few instances of grammatical or spelling updates remain in our dataset. Advanced models like GPT-4 could further reduce such cases.

4.2 When does INITIAL accurately answer NEW and UPDATED questions?

Although INITIAL was trained on Wikipedia's February 2023 data, it can answer questions about newer or revised information. This may be due to several reasons. First, some questions contain the answers within them, allowing correct responses without updated knowledge. For example, as shown in a sample from NEW04 in Figure 7, the answer is already in the question. Second, predictions can be made based on previous knowledge. For instance, in a sample from NEW03, certain answers might be inferred from keywords like "France" and "president". Lastly, the T5 pre-training, which includes various sources beyond Wikipedia, might have provided the model with relevant background knowl-
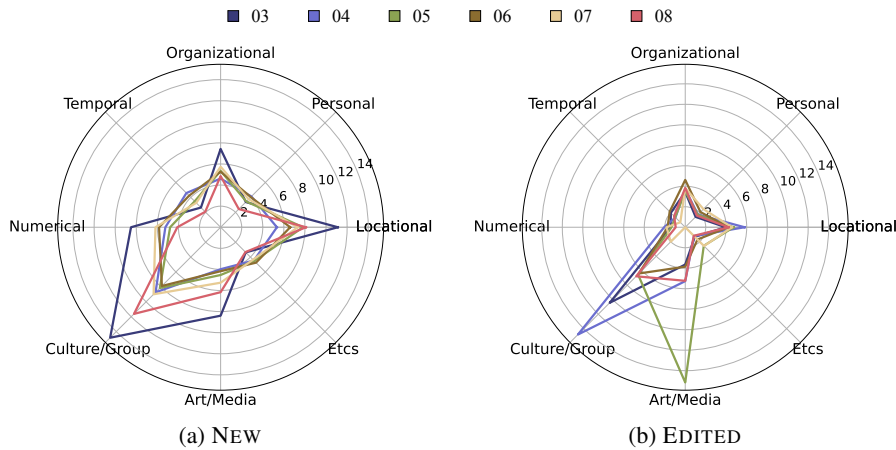
5407

(a) New   (b) Edited

Figure 6: The analysis of EM score according to QA category. The result of each time step is shown in different colors.

NEW03	
Question	Who has proposed to scrap the television licence fee in **France** and fund it directly from **the French** Treasury if re-elected as **president in 2022?**
Answer	**Emmanuel Macron**
Prediction	**Emmanuel Macron**
NEW04	
Question	What is the name of the railway **station** in **Niimi** where three JR West lines meet?
Answer	**Niimi station**
Prediction	**Niimi station**

Figure 7: Samples that INITIAL answers correctly from questions in NEW.

edge. The effectiveness of INITIAL in handling these questions also relates to the question difficulty level, our benchmark includes questions that are too easy to answer. This limitation can be improved with more advanced models like GPT-4 or by adjusting the question difficulty settings.

4.3 Can Retrieval Replace Continual Learning?

Section 3.3 reveals that DPR outperforms CL baselines in performance. However, this doesn't necessarily undermine the value of continual learning on language model. In EvolvingQA, it is easy for retrieval methods to search for relevant context because the questions often repeat words from their context, leading to a high overlap. For instance, the question "How many states have accepted the Affordable Care Act Medicaid extension?" directly mirrors its context's phrasing, when the context is "...39 states have accepted the Affordable Care Act Medicaid extension...". Additionally, since the questions merely seek specific facts, the model simply reads and identifies the apparent answer in the

context (i.e., executing one-hop prediction).

Continual learning remains crucial for language models, especially for real-world applications requiring complex reasoning and deep subject understanding. Language models need to integrate and apply their intrinsic knowledge to complex tasks, a capability beyond the scope of retrieval methods. Future research can focus on creating CL benchmarks that evaluate language models' ability to logically process and update knowledge.

4.4 Is Closed-Book QA the Best Way to Assess the Knowledge of Model?

In our research, we utilize the closed-book QA (CBQA) task to assess the knowledge of models. This method, however, requires careful consideration to determine its effectiveness in assessing a model's knowledge. For instance, there's a distinction between what a language model knows and how it responds, implying that CBQA results may not fully capture a model's inherent knowledge (Lewis et al., 2020; Jiang et al., 2020). Lastly, the current evaluation metrics, EM and F1, relies on lexical matching, and has limitations in verifying the accuracy of the model's predictions (Jiang et al., 2020; Risch et al., 2021; Bulian et al., 2022; Kamalloo et al., 2023). While our work is in early stages of research on continual learning for language models, we anticipate that considering such factors will enable the creation of benchmarks that are closer to optimal in the future. This direction is left as a promising avenue for future research.

5 Related Works

Temporal Continual Learning Benchmarks in NLP Zhang and Choi (2021) and Kasai et al. (2023) introduced QA datasets for temporal or geographical adaptation, but require manual annotation and disregard continual learning scenario. Jang et al. (2022) constructed benchmark to reflect Wikipedia's dynamically changing knowledge in an automated manner, but they did not include an evaluation setting to measure updating outdated knowledge. Jang et al. (2021) and Liška et al. (2022) proposed CL benchmarks relying on expert annotation and filtering, resulting in few timestamps and remaining static from the time it was created.

Continual Learning and Model Editing There is an increasing interest in continual learning for language models, particularly focusing on domain-incremental and task-incremental learning (Chen et al., 2020; Qin et al., 2022; Dhingra et al., 2022; Razdaibiedina et al., 2023; Chen et al., 2023; Cole et al., 2023). However, the area of temporally evolving CL and the term of forgetting outdated knowledge remains relatively under-explored. In the context of model editing (De Cao et al., 2021; Mitchell et al., 2021; Meng et al., 2022a,b; Huang et al., 2023), which is primarily aimed at updating and rectifying errors in language models, existing research often overlooks scenarios involving sequential updates. Moreover, the focus predominantly remains on updating extant knowledge, with less attention given to the acquisition of entirely new information. More detailed related works are available in Appendix H.

6 Conclusion

Our research shed light on the importance for LMs capability of dynamically accumulating and revising information to reflect the continual evolution of world knowledge, which were under-explored in previous studies. Our proposed EvolvingQA benchmark includes evaluation for the adaptability of LLMs to such continual changes, revealing significant deficiencies in current models' abilities to forget and update outdated knowledge, especially in numerical and temporal data. Our findings show that this is due to the ineffectiveness of gradient update in managing updated knowledge. We hope that our work acts as a cornerstone for future research aiming to bridge the existing gaps in LLMs' temporal adaptation capabilities.

Limitation Our study's limitations include the EvolvingQA dataset's lack of real-time updates. As Wikipedia updates monthly, there's a gap between current events and their reflection in the dataset. Additionally, using a single LLM for dataset construction and filtering processes introduce noise. LLMs can hallucinate and generate inaccurate data, and validation using the same LLM may not be possible to completely eliminate such risks. Though usage of advanced models such as GPT-4 and different validation model may mitigate this, it remains a concern. Furthermore, the overall performance is low, since closed-book QA itself is a very challenging task, and this can be alleviated by training models with larger capacities (Roberts et al., 2020). Finally, our framework do not allow control over question difficulty, affecting the evaluation results depending on the complexity of the questions. This might be addressed with refined prompting or additional pre-processing.

Ethics Statement In the development and evaluation of our benchmark, we adhered to rigorous ethical standards concerning the use of data and the potential impacts of our research. Our approach to continual pre-training and knowledge updating was designed to avoid the perpetuation of temporal biases, inaccuracies, or outdated information. We acknowledge that our benchmark and language models trained on it can be susceptible to reflecting societal biases present in training data. We will make every effort and take all possible measures to minimize and avoid such risks to the best of our ability.

References

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.

Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Boerschinger, and Tal Schuster. 2022.

Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. *arXiv preprint arXiv:2202.07654*.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. *arXiv preprint arXiv:2004.12651*.

Wuyang Chen, Yanqi Zhou, Nan Du, Yanping Huang, James Laudon, Zhifeng Chen, and Claire Cui. 2023. Lifelong language pretraining with distribution-specialized experts. In *International Conference on Machine Learning*, pages 5383–5395. PMLR.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Jeremy R Cole, Aditi Chaudhary, Bhuwan Dhingra, and Partha Talukdar. 2023. Salient span masking for temporal understanding. *arXiv preprint arXiv:2303.12860*.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. *arXiv preprint arXiv:2104.08164*.

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.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. 2023. Transformer-patcher: One mistake worth one neuron. *arXiv preprint arXiv:2301.09785*.

Steven CY Hung, Jia-Hong Lee, Timmy ST Wan, Chein-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. 2019. Increasingly packing multiple facial-informatics modules in a unified deep-learning model via lifelong learning. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, pages 339–343.

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. *arXiv preprint arXiv:2204.14211*.

Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Stanley Jungkyu Choi, and Minjoon Seo. 2021. Towards continual knowledge learning of language models. *arXiv preprint arXiv:2110.03215*.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2021. Lifelong pretraining: Continually adapting language models to emerging corpora. *arXiv preprint arXiv:2110.08534*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Ehsan Kamalloo, Nouha Dziri, Charles LA Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. *arXiv preprint arXiv:2305.06984*.

Haeyong Kang, Jaehong Yoon, Sultan Rizky Hikmawan Madjid, Sung Ju Hwang, and Chang D Yoo. 2022. On the soft-subnetwork for few-shot class incremental learning. *arXiv preprint arXiv:2209.07529*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

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

Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. 2017a. Overcoming catastrophic forgetting by incremental moment matching.

Sang-Woo Lee, Jin-Hwa Kim, Jaehyun Jun, Jung-Woo Ha, and Byoung-Tak Zhang. 2017b. Overcoming catastrophic forgetting by incremental moment matching. *Advances in neural information processing systems*, 30.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2020. Question and answer test-train overlap in open-domain question answering datasets. *arXiv preprint arXiv:2008.02637*.

Zhizhong Li and Derek Hoiem. 2016a. Learning without forgetting.

Zhizhong Li and Derek Hoiem. 2016b. Learning without forgetting. In *14th European Conference on Computer Vision, ECCV 2016*, pages 614–629. Springer.

Adam Liška, Tomáš Kočiský, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien de Masson d'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, et al. 2022. Streamingqa: a benchmark for adaptation to new knowledge over time in question answering models. *arXiv preprint arXiv:2205.11388*.

Arun Mallya, Dillon Davis, and Svetlana Lazebnik. 2018. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 67–82.

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 Andonian, Yonatan Belinkov, and David Bau. 2022b. Massediting memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2021. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*.

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

Sumit Neelam, Udit Sharma, Hima Karanam, Shajith Ikbal, Pavan Kapanipathi, Ibrahim Abdelaziz, Nandana Mihindukulasooriya, Young-Suk Lee, Santosh Srivastava, Cezar Pendus, et al. 2022. A benchmark for generalizable and interpretable temporal question answering over knowledge bases. *arXiv preprint arXiv:2201.05793*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Yujia Qin, Jiajie Zhang, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2022. Elle: Efficient lifelong pre-training for emerging data. *arXiv preprint arXiv:2203.06311*.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

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.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. 2023. Progressive prompts: Continual learning for language models. *arXiv preprint arXiv:2301.12314*.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010.

Julian Risch, Timo Möller, Julian Gutsch, and Malte Pietsch. 2021. Semantic answer similarity for evaluating question answering models. *arXiv preprint arXiv:2108.06130*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. *Advances in Neural Information Processing Systems*, 32.

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. 2016. Progressive neural networks. *arXiv preprint arXiv:1606.04671*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. 2017. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30.

Sebastian Thrun. 1995. *A Lifelong Learning Perspective for Mobile Robot Control*. Elsevier.

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

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149.

Jaehong Yoon, Sung Ju Hwang, and Yue Cao. 2023. Continual learners are incremental model generalizers. In *International Conference on Machine Learning*.

Michael J. Q. Zhang and Eunsol Choi. 2021. Situatedqa: Incorporating extra-linguistic contexts into qa.

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

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2023b. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*.

Zexuan Zhong, Zhengxuan Wu, Christopher D Manning, Christopher Potts, and Danqi Chen. 2023. Mquake: Assessing knowledge editing in language models via multi-hop questions. *arXiv preprint arXiv:2305.14795*.

A   Dataset Details

We collect Wikipedia snapshot from February 2023 to August 2023. For snapshot from February 2023, we use entire articles to pre-train INITIAL. We extract changes from two consecutive snapshots, and filter out articles that do not include much edited parts. The number of articles from each process is reported in Table 4.

To create CHANGED sets from these resulting articles, we use salient span masking, and a sample from CHANGED03 set is shown in Figure 8. In the input, named entities and dates are masked, and the output contains the masked entities.

B   Comparison of EvolvingQA with Other Benchmarks

Table 1 reports the comparison between EvolvingQA and the existing benchmarks for temporal alignment. EDITED KNOWLEDGE denotes evaluation on updated and outdated knowledge, and AUTOMATIC CONSTRUCTION denotes benchmark construction can be automated without human annotation. # OF TIME STEPS shows available time steps of the benchmark, while (Unlimited) denotes whether the construction framework can be applied dynamically to future time steps. AVAILABLE TASKS shows benchmark's downstream task. Our benchmark have significant advantages including evaluation of edited knowledge, ability to be constructed automatically with unlimited number of time steps, and question answering as practical downstream task.

C   EDITED Construction Details

In this section, we provide details on construction of EvolvingQA, especially about constructing EDITED set. Note that constructing EDITED requires a lot of filtering and refinement process, since Wikipedia update includes grammar and error correction, so the update may not include factual update. Therefore, we go through multiple process of filtering and extensive prompt engineering to obtain QA pairs that actually reflect factual update.

The prompt used to generate EDITED is described in Figure 3. Note that [System][4], [Assistant], and [User] indicate "role" when providing messages to GPT-3.5 through API. Below are the examples of prompts we use in every step of construction pipeline when validating EDITED set.

---

[4]The system message is used to control the behavior of the AI model, such as by providing specific instructions.

Time step (Month, 2023)	03	04	05	06	07	08
Entire snapshot	16,887,309	16,918,791	16,966,779	16,997,214	17,108,808	17,233,540
CHANGED w/o filtering	337,868	353,934	357,598	362,606	347,970	361,699
CHANGED	61,176	65,780	64,140	66,938	63,946	68,075

Table 4: The number of articles in CHANGED sets.

Changed03	
Input	2023 Memphis Tigers football team. The <extra_id_0> <extra_id_1> football team represented <extra_id_2> in the <extra_id_3> <extra_id_4> football season. The <extra_id_5> played their home games at <extra_id_6> in <extra_id_7>, <extra_id_8>, and competed in <extra_id_9> (The <extra_id_10>). They were led by <extra_id_11> head coach <extra_id_12>.<extra_id_13>.The <extra_id_14> finished <extra_id_15> <extra_id_16>, <extra_id_17> in <extra_id_18> play to finish in last <extra_id_19> place in the conference. The <extra_id_20> beat <extra_id_21> <extra_id_22>-10 in <extra_id_23>. Schedule.<extra_id_24> and <extra_id_25> (<extra_id_26>) announced the <extra_id_27> football schedule on <extra_id_28>.
Answer	<extra_id_0> 2023 <extra_id_1> Memphis Tigers <extra_id_2> the University of Memphis <extra_id_3> 2023 <extra_id_4> NCAA Division I FBS <extra_id_5> Tigers <extra_id_6> Liberty Bowl Memorial Stadium <extra_id_7> Memphis <extra_id_8> Tennessee <extra_id_9> the American Athletic Conference <extra_id_10> American <extra_id_11> fourth-year <extra_id_12> Ryan Silverfield <extra_id_13> Previous season <extra_id_14> Tigers <extra_id_15> the 2022 season <extra_id_16> 7–6 <extra_id_17> 3–5 <extra_id_18> Sun Belt <extra_id_19> eight <extra_id_20> Tigers <extra_id_21> Utah State <extra_id_22> 38 <extra_id_23> the First Responder Bowl <extra_id_24> Memphis <extra_id_25> the American Athletic Conference <extra_id_26> AAC <extra_id_27> 2023 <extra_id_28> February 21, 2023 <extra_id_29>

Figure 8: A sample of input and output from CHANGED03.

C.1 Filtering No Factual Update

The extracted QA instances still includes a number of instances that the outdated answer and the updated answer are written different, but actually the same. To filer out these cases, we prompt as below:

> Are '28' and 'Twenty-Eight' semantically equivalent or meaning the same?
> Options:
> (A) True
> (B) False
> Answer:

For above example, GPT-3.5 reponses as (A) True, then we filter out this instance from the dataset. There may be potential noise in our filtering process when using multiple-choice prompts (Zheng et al., 2023b). We incorporate varied seeds and altering the order of options.

C.2 Filtering Hallucination

For some instances, GPT-3.5 make up question even though there are no sufficient information in the context that supports the question and answer. In this regard, to filter out hallucinated instances, we use prompt following Kadavath et al. (2022):

> "Context of 'Commuter rail': Indonesia, the Metro Surabaya Commuter Line, Prambanan Express, KRL Commuterline Yogyakarta, Kedung Sepur, the Greater Bandung Commuter
> Question: Which commuter rail system was removed from the list in April 2023?
> Proposed Answer: the Greater Bandung Commuter
> Given the context, is the proposed answer:
> (A) True
> (B) False
> The proposed answer is:"

For the above case, GPT-3.5 responded (B) False, then we excluded this instance from the dataset.

D Evaluation on EDITED Knowledge in Multiple Choice Setting

Method	Knowledge	03	04	05	06	07	08
INITIAL	OUTDATED	53.33	53.04	52.37	53.1	54.49	53.52
	UPDATED	46.67	46.96	47.63	46.9	45.51	46.48
FULL	OUTDATED	52.21	51.94	51.61	50.78	53.41	52.4
	UPDATED	47.79	48.06	48.39	49.22	46.59	47.6
K-Adapter	OUTDATED	52.08	51.11	49.73	51.13	54.08	51.69
	UPDATED	47.92	48.89	50.27	48.87	45.92	48.31
LoRA	OUTDATED	52.07	50.59	50.94	51.13	53.87	52.4
	UPDATED	47.93	49.41	49.06	48.87	46.13	47.6

Table 5: The results of multiple choice setting on EDITED knowledge according to baseline methods.

Following previous studies (Brown et al., 2020; Sanh et al., 2021), we evaluate the baselines on EDITED knowledge using multiple choice setting (i.e., rank classification), which is selecting the label option (i.e., either outdated or updated) with higher log-likelihood. Namely, the model computes the log probability of both updated and outdated ground-truth answer and uses the higher one as the predicted answer. The log proability is calculated by summing the negative log softmax logits of the model on the tokens in ground-truth answer. The result reported in Table 5 shows that all the baselines fail to capture updated knowledge, and tend to be skewed more to outdated knowledge.

E Prompting Time Information

We add time information in the question, to see how e language model answers updated knowledge

|(a) UNCHANGED|(b) EDITED|(c) NEW|

Figure 9: Comparison between with and without adding time information into questions. The darker color indicates the result of adding time information. The EM score is averaged for all time steps.

correctly after conditioning on time information. Specifically, when we test our models trained on CHANGED05, we then prepend "As of May 2023," to all the questions in UNCHANGED05, NEW05, and EDITED05. The result in Figure 9 shows that inserting time information deteriorates the performance significantly. This is in line with Kasai et al. (2023) that in closed-book QA task, their date insertion method does not improve the performance. When we analyze the model's prediction when time information is given, the models tend to hallucinate more on temporal questions. Namely, when the models are asked to answer temporal questions asking dates, the models tend to reply with the date given as time information.

F   Additional Results on Gradient Analysis of EDITED

Figure shows additional result of gradient norm analysis on CHANGED04. As in Section 3.4.1, the result shows that gradient norm when learning edited knowledge is generally smaller than new knowledge. Note that we use instances from CHANGED04 set using checkpoint from FULL after pre-trained on CHANGED03 and calculate gradients of the entire parameters.
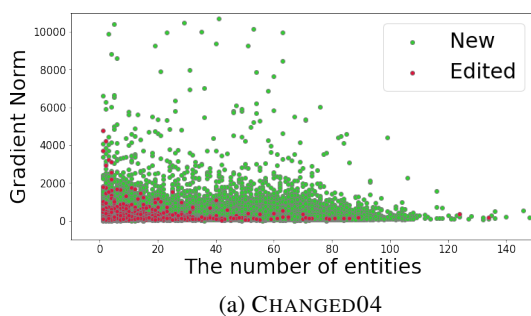

(a) CHANGED04

Figure 10: The scatter plot of samples from CHANGED04 according to the number of masked entities and gradient norm. Each dot indicates a sample from either NEW knowledge or EDITED knowledge in CHANGED. The x-axis shows the Frobenius norm of weight gradients of each sample. The y-axis shows the number of masked entities in a sample.

G   Details about Categorization of QA samples

For categorization, we employ a Named Entity Recognition (NER) model to classify the categories of answers in our benchmark. The 'Numerical' category encompasses answers identified as cardinal or ordinal numbers, quantities, and percentages. The 'Temporal' category includes dates and times, while 'Locational' encompasses geopolitical or geographical locations and facilities. 'Organizational' refers to entities like organizations, and 'Culture/Group' includes languages, laws, nationalities, and religious or political groups. 'Art/Media' covers events, works of art, and products. Finally, 'Etcs' comprises answers that do not fit into the other categories.

H   Additional Related Works

Continual Learning Continual learning (CL) is often categorized in three directions: *Regularization-based* approaches (Li and Hoiem, 2016b; Lee et al., 2017b; Yoon et al., 2023) aim to regularize the changes of model parameters to avoid forgetting previous knowledge during continual learning; *Architecture-based* approaches (Rusu et al., 2016; Mallya et al., 2018; Hung et al., 2019; Kang et al., 2022) utilize different parameters or modules for each task to prevent forgetting; and *Replay-based* approaches (Rebuffi et al., 2017; Shin et al., 2017; Rolnick et al., 2019) store a subset of training samples or other useful data in a replay buffer and learn new tasks by referring to the buffer.

Along with the remarkable advances in vision-based continual learning, the importance of continual learning for language models has been recognized in recent days (Chen et al., 2020; Qin et al., 2022; Dhingra et al., 2022; Razdaibiedina et al., 2023; Chen et al., 2023; Cole et al., 2023). However, most of these works focus on domain-incremental CL, which continually learn different domain corpora such as bio-medical papers to physics papers (Jin et al., 2021; Qin et al., 2022), or task-incremental CL (Chen et al., 2020; Razdaibiedina et al., 2023). However, research on temporal evolving continual learning is yet under-explored.

Model Editing Model editing is proposed to keep language models up-to-date and fix any errors in their existing knowledge. There are four

main approaches for model editing. Memory-based approaches (Mitchell et al., 2022; Zhong et al., 2023; Zheng et al., 2023a) retrieve the most relevant edit facts from external memory. Parameter-expansion approaches (Huang et al., 2023) train additional parameters with modified knowledge. Locate-then-edit approaches (Meng et al., 2022a,b) identify the specific parts of the model that need changes and updates them directly. Meta-learning based approaches (De Cao et al., 2021; Mitchell et al., 2021) employ a hyper-network trained to predict the necessary gradient update for editing. However, model editing studies overlook multiple updates scenario (i.e., more than 2 update steps), or focus only on knowledge update, disregarding knowledge addition. Moreover, they update knowledge in fine-tuning stage, but continual learning learns and update knowledge during continual pre-training, which enables large amount of knowledge update and close to real-world scenario.