Event-level Knowledge Editing

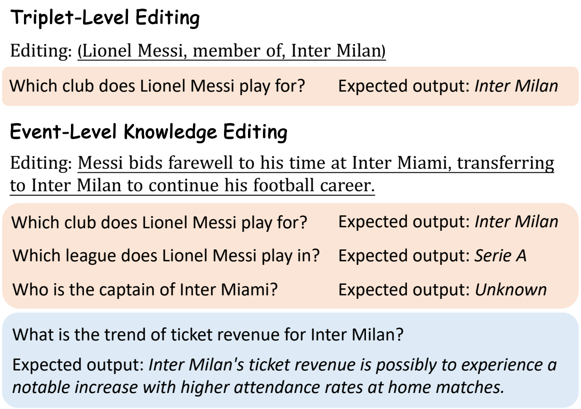
Anonymous ACL submission

Abstract

Knowledge editing aims at updating knowledge of large language models (LLMs) to prevent them from becoming outdated. Existing work edits LLMs at the level of factual knowledge triplets. However, natural knowledge updates in the real world come from the occurrences of new events rather than direct changes in factual triplets. In this paper, we propose a new task setting: **event-level knowledge editing**, which directly edits new events into LLMs and improves over conventional triplet-level editing on (1) **Efficiency**. A single event edit leads to updates in multiple entailed knowledge triplets. (2) **Completeness**. Beyond updating factual knowledge, event-level editing also requires considering the event influences and updating LLMs' knowledge about future trends. We construct a high-quality event-level editing benchmark ELKEN, consisting of $1,515$ event edits, $6,449$ questions about factual knowledge, and $10,150$ questions about future tendencies. We systematically evaluate the performance of various knowledge editing approaches and LLMs on this benchmark. We find that ELKEN poses significant challenges to existing knowledge editing approaches. Our codes and dataset are publicly released to facilitate further research.

1 Introduction

The world is constantly evolving, with new knowledge emerging frequently, leading to outdated or even misleading knowledge within language models (LLMs). Therefore, numerous works focus on knowledge editing, aiming to update new knowledge into LLMs. (Sinitsin et al., 2019; De Cao et al., 2021; Meng et al., 2022a,b; Mitchell et al., 2022; Wang et al., 2023a; Zheng et al., 2023; Zhang et al., 2024). Previous work defines knowledge editing as triplet-level editing, which edits factual knowledge triples into LLMs. As shown in Figure 1, supposing the triplet-level editing updates a new factual triplet (`Lionel Messi, member of, Inter Milan`) into LLMs, the model's answer to "*Which club does Lionel Messi play for?*" should be changed to `Inter Milan`.



Triplet-Level Editing

Editing: (Lionel Messi, member of, Inter Milan)

Which club does Lionel Messi play for? Expected output: *Inter Milan*

Event-Level Knowledge Editing

Editing: Messi bids farewell to his time at Inter Miami, transferring to Inter Milan to continue his football career.

Which club does Lionel Messi play for? Expected output: *Inter Milan*

Which league does Lionel Messi play in? Expected output: *Serie A*

Who is the captain of Inter Miami? Expected output: *Unknown*

What is the trend of ticket revenue for Inter Milan?

Expected output: *Inter Milan's ticket revenue is possibly to experience a notable increase with higher attendance rates at home matches.*

Figure 1: A counterfactual example for triplet-level and event-level knowledge editing. Triplet-level editing updates factual triplets into models. Event-level editing updates events into models, thus efficiently modifying factual knowledge and tendencies of models.

However, triplet-level editing is unnatural, as knowledge updates in the real world happen with new events rather than direct updates to knowledge triples. For example, in Figure 1, the update of the knowledge (`Lionel Messi, member of, Inter Milan`) is due to the event that Lionel Messi transfers to Inter Milan. Moreover, triplet-level editing has the following limitations: (1) **Inefficiency**. An event may update multiple factual triplets at once. As in Figure 1, Messi's transfer to Inter Milan updates several facts, including the sports club of Messi, the league where Messi plays, and Inter Miami's captain, etc. When a new event occurs, triplet-level editing needs to identify all affected triplets in advance before editing, which is time-consuming and labor-intensive. (2) **Incompleteness**. An event not only updates definite factual knowledge but can also affect potential tendencies of the future. For example, in Figure 1, Messi's transfer to Inter Milan could influence the tendency of ticket revenue for Inter Milan. Updating tendency knowledge in LLMs is crucial for enabling more reliable responses, such as event forecasting (Zou et al., 2022; Halawi et al., 2024). However, existing triplet-level editing ignores the update in tendency knowledge.

Given the above issues, we propose a new task setting, **event-level knowledge editing**, aimed at editing newly occurred events into LLMs, thereby updating multiple factual knowledge and influenced tendencies at once. Event-level knowledge editing addresses the above limitations in two aspects: (1) **Updating all implicated facts at once**. Unlike triplet-level editing which requires explicitly identifying all the influenced triplets before editing, event-level editing aims at updating all the implicated factual triplets with a single event edit. For instance in Figure 1, after editing the event of Messi's transfer to Inter Milan into LLMs, the models should modify its multiple factual knowledge, such as the sports club of Messi, the league Messi plays in, and Messi's work location. This requires the model to infer all the factual triplets influenced by the event and also involves *multi-hop reasoning* (Zhong et al., 2023), such as the update of the league where Messi plays due to Messi playing for Inter Milan and Inter Milan being a club of the Serie A league. Furthermore, we also consider the scenario of editing knowledge to *unknown* (Muresanu et al., 2024), which has not been explored to our knowledge. For example, in Figure 1, since Messi is no longer the captain of Inter Miami, and without additional information, Inter Miami's captain should be edited to *unknown*. (2) **Updating tendency knowledge**. Beyond definite factual knowledge, event-level knowledge editing also enables updating the uncertain knowledge about future trends considering the new events. For example, in Figure 1, after editing the event of Messi's transfer to Inter Milan into LLMs, the models should adjust their knowledge on some tendencies, such as the tendency of ticket revenue for Inter Milan. This requires the model to understand the broad impact of event editing and possess *common sense knowledge* (Gupta et al., 2023). For instance, in Figure 1, correctly predicting the tendency of ticket revenue for Inter Milan necessitates knowing that Messi is a football superstar and will draw more fans to watch Inter Milan's matches.

We construct a high-quality benchmark ELKEN for event-level knowledge editing, including $1,515$ event edits along with $6,449$ questions for factual knowledge and $10,150$ questions for tendencies. To reduce costs and ensure that the construction methodology applies to other scenarios, we design a semi-automatic construction process. For factual knowledge, we manually create several event templates and their impacted triplets. We sample entities from Wikidata (Vrandečić and Krötzsch, 2014) to instantiate the templates and obtain event edits and question-answer pairs. We then use GPT-3.5 (OpenAI, 2022) to paraphrase the event edits to get the final diverse edits. For tendencies, we first reuse event edits generated for factual knowledge and augment them with events having a broader impact. We use GPT-3.5 to generate tendency-related question-answer pairs and verify the generated data with human annotation.

We conduct systematic experiments and analysis on ELKEN, evaluating 5 representative methods, including Fine-tuning (Yao et al., 2023), Spare and Dense Retrieval (Akyürek et al., 2023), SERAC (Mitchell et al., 2022), and In-Context Editing (ICE) (Akyürek et al., 2023), and 6 language models, including GPT-J (Wang and Komatsuzak, 2021), TULU 2 (Ivison et al., 2023), Mistral 7B (Jiang et al., 2023), GPT-3.5 (OpenAI, 2022), GPT-4 (OpenAI, 2023), and Gemini Pro (Team et al., 2023). We find that the event-level knowledge editing task presents significant challenges to existing editing methods and models, which highlights the importance of future research.

2 Event-level Knowledge Editing

2.1 Task Definition

Event-level knowledge editing aims to edit events into LLMs, thereby updating both influenced definite factual knowledge and uncertain knowledge about future tendencies at once. The objectives and challenges of event-level knowledge editing primarily include two aspects: (1) **Updating all implicated facts at once**. An event edit can update multiple factual knowledge at once, and determining its scope is challenging. Additionally, updating corresponding factual knowledge about an event edit may involve *multi-hop reasoning* (Zhong et al., 2023) and editing knowledge to *unknown* (Muresanu et al., 2024). (2) **Updating tendency knowledge**. An event edit can also update uncertain knowledge about future tendencies, and identifying the broad tendency impacts of an event edit is challenging, usually requiring common sense knowledge (Gupta et al., 2023).

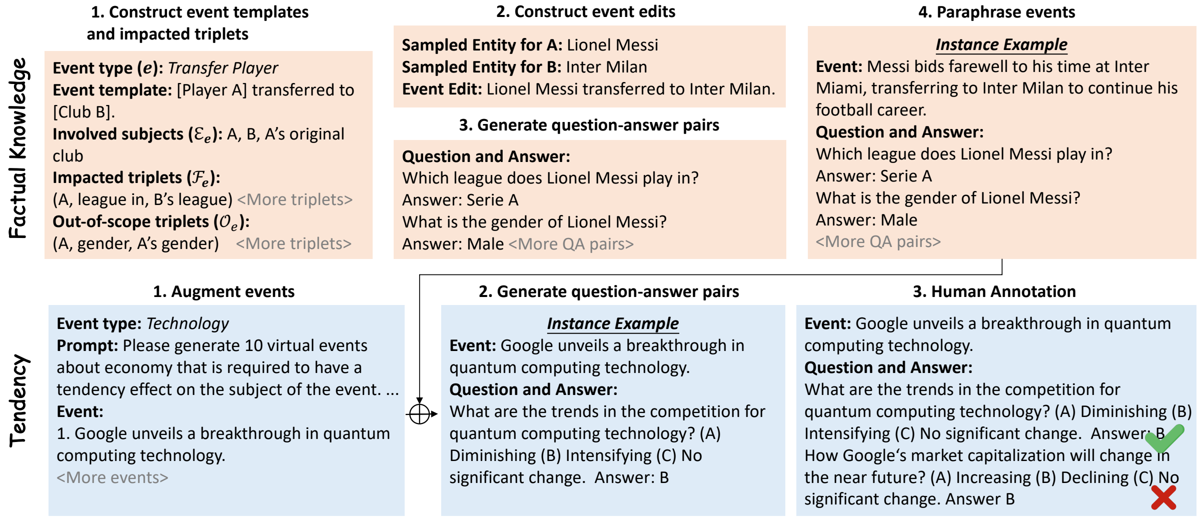

Factual Knowledge
1. Construct event templates and impacted triplets
Event type (e): Transfer Player
Event template: [Player A] transferred to [Club B].
Involved subjects ($\mathcal{E}_e$): A, B, A's original club
Impacted triplets ($\mathcal{F}_e$):
(A, league in, B's league) <More triplets>
Out-of-scope triplets ($\mathcal{O}_e$):
(A, gender, A's gender) <More triplets>
2. Construct event edits
Sampled Entity for A: Lionel Messi
Sampled Entity for B: Inter Milan
Event Edit: Lionel Messi transferred to Inter Milan.
3. Generate question-answer pairs
Question and Answer:
Which league does Lionel Messi play in?
Answer: Serie A
What is the gender of Lionel Messi?
Answer: Male <More QA pairs>
4. Paraphrase events
Instance Example
Event: Messi bids farewell to his time at Inter Miami, transferring to Inter Milan to continue his football career.
Question and Answer:
Which league does Lionel Messi play in?
Answer: Serie A
What is the gender of Lionel Messi?
Answer: Male
<More QA pairs>
Tendency
1. Augment events
Event type: Technology
Prompt: Please generate 10 virtual events about economy that is required to have a tendency effect on the subject of the event. ...
Event:
1. Google unveils a breakthrough in quantum computing technology.
<More events>
2. Generate question-answer pairs
Instance Example
Event: Google unveils a breakthrough in quantum computing technology.
Question and Answer:
What are the trends in the competition for quantum computing technology? (A) Diminishing (B) Intensifying (C) No significant change. Answer: B
3. Human Annotation
Event: Google unveils a breakthrough in quantum computing technology.
Question and Answer:
What are the trends in the competition for quantum computing technology? (A) Diminishing (B) Intensifying (C) No significant change. Answer: B ✓
How Google's market capitalization will change in the near future? (A) Increasing (B) Declining (C) No significant change. Answer B ✗

Figure 2: The overall construction process of ELKEN, including two categories of question-answer pairs: `Factual Knowledge` and `Tendency`. *Instance Example* demonstrates a sample of the data.

Formally, given an event edit e, f_θ represents the model before the edit, with θ denoting the model's parameters, and f_{θ_e} denotes the model after editing the edit e. $\mathcal{F}_e$ and $\mathcal{T}_e$ represent the scope of factual knowledge and tendency impacted by e, respectively. We refer to the questions in $\mathcal{F}_e \cup \mathcal{T}_e$ as **in-scope** questions. Moreover, the editing process should not affect the model's unrelated knowledge (Yao et al., 2023), which are referred to as **out-of-scope** knowledge and denoted as $\mathcal{O}_e$. The goal of event-level knowledge editing is as follows:

$$f_{\theta_e}(x) = \begin{cases} y_e & x \in \mathcal{F}_e \cup \mathcal{T}_e \\ f_\theta(x) & x \in \mathcal{O}_e \end{cases} \tag{1}$$

y_e is the expected answer after editing. Based on this objective of event-level knowledge editing, we assess the editing methods from two dimensions: reliability and locality.

Reliability assesses whether the edited model answers as expected, evaluating the accuracy of answers to in-scope questions about $\mathcal{F}_e \cup \mathcal{T}_e$:

$$\mathbb{E}_{(x,y_e)\in\mathcal{F}_e\cup\mathcal{T}_e}\mathbb{1}\{\text{argmax}_y f_{\theta_e}(y|x) = y_e\}. \tag{2}$$

Locality means that the editing should not affect the model's answers to unrelated questions, evaluating the consistency of the model's answers to the unrelated questions in $\mathcal{O}_e$ before and after editing:

$$\mathbb{E}_{(x,y_e)\in\mathcal{O}_e}\mathbb{1}\{f_{\theta_e}(y|x) = f_\theta(y|x)\}. \tag{3}$$

2.2 Benchmark Construction

Our ELKEN benchmark consists of data for factual knowledge impacts (`Factual Knowledge`) and tendency impacts (`Tendency`). Figure 2 illustrates the overall data construction process, and Table 1 shows the data statistics. More construction details and comparisons to existing triplet-level editing datasets are shown in appendix A.

Construction of `Factual Knowledge` Unlike the data construction of triplet-level editing, which only requires replacing entities within triplets for constructing edits and question-answer pairs (Yao et al., 2023), the construction of event-level editing is more complex, as identifying the impact scope of an event is difficult.

To this end, we propose a semi-automatic approach that conserves human efforts while ensuring data quality and is transferable to other scenarios. As illustrated in Figure 2, the overall construction process of `Factual Knowledge` consists of 4 steps. (1) **Constructing event templates and their impacted triplets**. Our method for determining the impact scope is similar to "Ripple Effects" (Cohen et al., 2023), involving manual efforts, but the impact scope of events is broader, involving more subjects and triplets. We first select 16 common event types from MAVEN (Wang et al., 2020, 2023b) and ACE 2005 (Walker et al., 2006) that are likely to lead to changes in factual knowledge. We also select 81 widely-used relationships from Wikidata, denoted as $\mathcal{R}$. For each event type, we manually construct an event template. For example, for the "*Transfer Player*" type, its template is "*[Player A] transferred to [Club B]*" with placeholders `A` and `B`. We then manually identify the directly involved subjects of the event e, denoted as $\mathcal{E}_e$. For each subject s in $\mathcal{E}_e$, we refer to $\mathcal{R}$ and manually identify the scope of triplets $\mathcal{F}_s$ of s impacted by this event, which consists of $(s, r, o*)$, where $r \in \mathcal{R}$ and $o*$ denotes the updated answer. For instance in Figure 2, $\mathcal{E}_e$ includes A, B, and A's original club. $\mathcal{F}_s$ of the subject A includes (A, member of, B), etc. We then aggregate $\mathcal{F}_s$ of each subject in $\mathcal{E}_e$ as $\mathcal{F}_e$. We adpot the same method to construct out-of-scope triplets $\mathcal{O}_e$, which consists of (s, r, o) where $s \in \mathcal{E}_e$, $r \in \mathcal{R} \setminus \{r|(s, r, o*) \in \mathcal{F}_e\}$, and o denotes the ground truth answer from Wikidata. To ensure a comprehensive and accurate identification of an event's impact scope, we involve 3 annotators to identify the impact scope, $\mathcal{F}_e$, and then we assemble all their annotations. (2) **Constructing event edits**. We instantiate event templates to create event edits. Specifically, we construct edits by sampling the 100 most frequent entities of corresponding types from Wikidata (Vrandečić and Krötzsch, 2014) based on the frequency counts from Wikipedia[1] to replace placeholders with specific entities. (3) **Generating question-answer pairs**. With the instantiated event edits and impacted triplets, we generate in-scope question-answer pairs. For each triplet $(s, r, o*)$ in $\mathcal{F}_e$, we adopt predefined rules to transform (s, r) as a question and take the instantiated $o*$ as the answer. For each triplet in $\mathcal{O}_e$, we adopt the same method to construct out-of-scope question-answer pairs. (4) **Paraphrasing event edits**. Finally, to make the expressions of event edits more natural and enhance linguistic diversity, we use LLMs to paraphrase the instantiated event templates and generate the final event edits. Specifically, we employ GPT-3.5 (OpenAI, 2022) in paraphrasing as GPT-3.5 is demonstrated as an effective paraphraser (Cegin et al., 2023). We manually review and verify each paraphrased event and find little noise. Finally, we obtain 841 event edits, 3,307 in-scope questions, and 3,142 out-of-scope questions in Factual Knowledge. We divide the data into a training set and a test set by event types.

		Train	Test
	#Event Edits	671	844
Factual Knowledge	#In-scope Q	971	2,171
	#Out-of-scope Q	1,325	1,982
Tendency	#In-scope Q	3,889	3,968
	#Out-of-scope Q	1,353	940

Table 1: Overall statistics of ELKEN. Q: Question. ELKEN comprises two categories of question-answer pairs: Factual Knowledge and Tendency. Tendency has two evaluation formats: Tendency-M for multiple choice and Tendency-G for open-ended generation.

Construction of Tendency Tendency reuses event edits from Factual Knowledge, which are usually specific and have limited broad impacts. To comprehensively evaluate LLMs' understanding of the broader tendency impact of events, we augment some event edits for Tendency by generating new events with LLMs. To identify the scope of impacted tendencies ($\mathcal{T}_e$), we preliminarily examine with human annotators and find manually crafted question-answer pairs about tendencies homogeneous and the process is labor-intensive. Previous work has shown that LLMs can provide reasonable predictions about future tendencies (Wang et al., 2023b; Halawi et al., 2024). Therefore, we adopt LLMs first to generate a rich set of question-answer pairs about tendencies, followed by manual annotation. Specifically, as shown in Figure 2, the construction process of Tendency includes 3 steps: (1) **Augmenting events**. We collect 18 event topics, such as *politics*, *sports*, etc., and use GPT-3.5 (OpenAI, 2022) to generate 100 counterfactual events for each event topic. We then filter out repeated events following Wang et al. (2023c). (2) **Generating question-answer pairs**. We use GPT-3.5 to generate tendency-related questions and answers. Although exhausting all possible tendencies is impossible, we prompt GPT-3.5 to generate rich and representative question-answer pairs through instructions and diverse demonstrations. For each event edit, we generate 6 in-scope and 2 out-of-scope question-answer pairs, each consisting of one question, three choices, and one answer. We manually assess 100 sampled questions and answers and find the accuracy rate is about 85% and the questions exhibit great diversity (covering much more topics than human-written questions), indicating the high quality of the model-generated data. (3) **Human Annotation**. Same as Factual Knowledge, we divide the data into a training set and a test set by event topics and types. To ensure the benchmark's quality, we manually annotate the whole test set to verify the model-generated questions and answers. To maintain annotation quality, all data are annotated twice and similar questions are filtered out. The final inter-annotator agreement reaches 95.6%. We filter out

[1] https://en.wikipedia.org

questions with inconsistent annotations and those whose answers are marked as incorrect. Finally, we obtain $1,515$ event edits, including 841 edits from `Factual Knowledge` and 674 newly generated edits, $7,857$ in-scope questions, and $2,293$ out-of-scope questions.

3 Experiments

3.1 Experimental Setup

As mentioned in § 2.1, the evaluation metrics have two dimensions: reliability and locality. For `Factual Knowledge`, the reliability is accuracy and the locality is the proportion of answers that are the same before and after editing. These metrics are calculated using **exact match**. For `Tendency`, the evaluation is more complicated since the tendency judgment is an open-ended generation task by nature, and we evaluate it under two settings. For the first generation setting (`Tendency-G`), we adopt an automated evaluation method using GPT-4 (OpenAI, 2023) as the evaluator, which has been verified as an effective (Bai et al., 2023; Li et al., 2024). Specifically, for the evaluation of reliability, we use the correct option of each question as the reference, comprehensively scoring the editing methods in 3 dimensions: *correctness*, *coherence*, and *comprehensiveness*. We also ask GPT-4 to give an *overall* score. Similar to the previous scoring-based evaluation method (Li et al., 2024), all scores are integers scaling from 1 to 5, with 5 being the best. For the evaluation of locality under `Tendency-G`, we utilize GPT-4 to assess the consistency of the model's responses to out-of-scope questions in $\mathcal{O}_e$ before and after editing, also using an integer score from 1 to 5, with 5 being the most similar. To avoid the potential risk of GPT-4 evaluation and provide an alternative metric for `Tendency`, we also adopt a **multiple-choice** evaluation setting (`Tendency-M`), which is the same with `Factual Knowledge`, i.e., using extract match to calculate the reliability and locality. Experimental details of using GPT-4 scorer on `Tendency-G` are shown in appendix B.2.

For reliability, we employ evaluations at two levels: *question-level* and *edit-level*. The *question-level* evaluation assesses the reliability of each individual question. For `Tendency-G` scores evaluated using GPT-4, similar to Bai et al. (2023), we present the percentages of responses with full marks, *i.e.*, scored 5 points. The *edit-level* evaluation assesses the reliability of each edit. An edit is reliable only if all questions in $\mathcal{F}_e$ (for `Factual Knowledge`) or $\mathcal{T}_e$ (for `Tendency-M` and `Tendency-G`) are answered correctly *or* the *overall* scores of answers are all full-mark.

3.2 Investigated Editing Methods and Models

We evaluate various advanced editing methods on our benchmark, including: (1) **Fine-tuning** (Zhu et al., 2020; Meng et al., 2022b; Akyürek et al., 2023). Fine-tuning is a vanilla method of editing, involving direct learning the new edits by fine-tuning model parameters. In our experiments, we fine-tune all the parameters of models on edits in the test set using a language modeling objective. However, this method has high computational costs and may also lead to catastrophic forgetting (Luo et al., 2023). (2) **Retrieval** (Madaan et al., 2022a; Zhong et al., 2023). This method is memory-based, which stores all edits in an external memory. When posed with a question, this method first retrieves the most matching edit to use as context along with the question for input into the models. In our experiments, we used BM25 and E5 (Wang et al., 2022) as the retrieval methods, named *sparse retrieval* and *dense retrieval*, respectively. (3) **SERAC** (Mitchell et al., 2022) trains a scope classifier to determine whether a question requires retrieving a corresponding edit for answering. If retrieval is necessary, the retrieved edit and the question are input together into a *counterfactual* model for answering; otherwise, the question alone is input into the vanilla pre-trained model. In our implementation, we train a cross-encoder classifier (Mitchell et al., 2022) based on ELECTRA (Clark et al., 2019) and we use the same pre-trained model as the *counterfactual* model, which is the same as in Akyürek et al. (2023). (4) **ICE**. This method inputs the ground truth edit as context along with the question into pre-trained models, directly evaluating whether the model can correctly answer questions. We do not evaluate some advanced approaches, such as the Locate-Then-Edit methods (Dai et al., 2022; Meng et al., 2022a,b; Li et al., 2023), because these approaches are specifically designed for triple-level editing and are not applicable to event-level knowledge editing.

We adopt several advanced language models as the base models to implement the aforementioned methods. We employ three open-source models, including GPT-J (Wang and Komatsuzak, 2021), TULU 2 (Ivison et al., 2023), Mistral 7B (Jiang et al., 2023), and three powerful proprietary models,

Model	Method	`Factual Knowledge`			`Tendency-M`			`Tendency-G`			E-Level
		E-Level	Q-Level	Locality	E-Level	Q-Level	Locality	E-Level	Q-Level	Locality	
GPT-J	Fine-tuning	0.0	2.3	67.2	4.8	41.3	99.3	0.2	4.8	87.7	0.1
	Sparse Retrieval	15.1	44.2	29.4	7.5	49.3	42.8	0.7	12.0	51.6	0.2
	Dense Retrieval	16.7	46.7	28.8	7.3	49.2	40.6	0.6	12.5	53.8	0.1
	SERAC	4.3	20.4	65.1	8.2	49.4	74.4	0.0	4.2	80.3	0.0
	ICE	17.1	49.9	29.1	7.5	49.6	41.6	1.6	13.7	54.3	0.9
TULU 2	Fine-tuning	0.0	4.7	**90.6**	10.3	53.9	**100.0**	5.1	36.3	81.3	2.9
	Sparse Retrieval	26.4	56.3	53.7	16.4	63.4	40.3	2.9	34.0	28.5	2.6
	Dense Retrieval	28.7	59.3	52.8	24.9	69.5	42.1	7.1	41.4	25.2	5.5
	SERAC	7.0	30.4	89.1	18.9	65.3	59.0	4.6	38.0	76.7	2.9
	ICE	30.5	63.8	53.7	34.1	75.9	39.3	9.5	43.8	25.1	8.6
Mistral 7B	Fine-tuning	0.2	4.1	66.8	21.5	65.4	**100.0**	19.1	59.7	77.6	10.5
	Sparse Retrieval	24.1	57.5	39.6	28.0	72.5	34.7	6.1	43.9	28.6	3.9
	Dense Retrieval	25.6	60.4	39.1	40.5	79.2	37.6	12.8	53.6	21.4	9.9
	SERAC	7.4	27.4	71.1	37.4	76.4	59.8	14.1	54.7	74.7	8.6
	ICE	26.6	64.5	39.8	60.1	88.0	35.6	21.5	59.6	22.8	16.7
GPT-3.5	Sparse Retrieval	16.9	55.0	33.0	49.4	82.4	41.2	10.4	48.7	23.9	7.2
	Dense Retrieval	18.4	60.2	30.6	57.6	86.0	46.6	21.5	60.1	19.3	16.3
	SERAC	5.2	27.1	71.2	56.0	84.9	70.3	17.9	57.7	70.6	11.7
	ICE	20.0	63.1	32.7	**71.6**	**91.6**	41.9	33.8	66.6	20.1	27.1
GPT-4	Sparse Retrieval	34.2	64.7	56.6	30.6	71.8	52.0	14.5	58.4	34.0	9.8
	Dense Retrieval	36.5	68.7	56.1	46.5	80.6	51.8	24.3	66.1	31.0	18.9
	SERAC	9.7	31.4	80.6	45.8	81.4	92.3	26.5	65.7	**93.1**	15.2
	ICE	**39.0**	**73.5**	56.9	66.4	89.3	49.8	**40.3**	**73.0**	31.9	**29.2**
Gemini Pro	Sparse Retrieval	24.3	60.3	30.3	13.8	57.3	38.0	2.8	29.8	33.7	2.4
	Dense Retrieval	25.2	63.7	30.5	28.2	67.2	43.4	6.6	39.5	33.9	5.6
	SERAC	6.0	28.3	72.1	31.1	72.3	77.0	8.4	45.7	70.3	5.5
	ICE	24.3	65.6	41.6	41.9	75.2	40.6	7.4	38.1	39.7	7.2

Table 2: Experimental results (%) of all investigated methods and models on ELKEN. E-Level: Edit-level reliability. Q-Level: Question-level reliability. The results on `Tendency-G` are the percentages of full-mark of *overall* scores. The rightmost column, E-Level, displays the overall reliability considering $\mathcal{F}_e \cup \mathcal{T}_e$.

including GPT-3.5 (OpenAI, 2022), GPT-4 (OpenAI, 2023), and Gemini Pro (Team et al., 2023). The implementation details of the editing method and automated evaluation are in appendix B.

3.3 Experimental Results

The experimental results are shown in Table 2, and we have the following general observations: (1) Existing methods exhibit moderate performance on ELKEN. Even the best-performing method (ICE + GPT-4) falls short, which indicates the significant challenge posed by the new event-level knowledge editing setting. (2) The question-level reliability scores on ELKEN are much lower than those in triplet-level editing. For instance, SERAC can achieve nearly 100% reliability (Yao et al., 2023) in triplet-level editing. Moreover, the reliability scores of event-level evaluations are further lower than those of question-level evaluations. This suggests that recognizing the impact scope of event editing is a novel challenge of our task. The impact scope of triplet-level editing typically confines to edited triplets themselves, while that of event-level knowledge editing extends to multiple factual and tendency knowledge. (3) The locality scores on ELKEN are also generally lower than those in triplet-level editing. For example, SERAC + GPT-J achieves nearly 100% locality in triplet-level editing (Yao et al., 2023) but only attains about 80% and 65.1% in `Tendency` and `Factual Knowledge`, respectively. This may be due to the broad impact range of event edits, making the models struggle to ensure the locality of edits, which poses new challenges to existing methods. (4) On in-scope questions of `Tendency-G`, the full-mark rate is lower compared to reliability scores on `Tendency-M`. This is because `Tendency-G` not only assesses the tendency correctness of answers but also evaluates the coherence and comprehensiveness. This indicates that although the model may correctly identify the tendency of a question, it struggles to provide comprehensive and reasonable explanations.

4 Further Analysis

This section presents some further analyses. Unless otherwise specified, the experimental results are from the ICE method, with *question level* reliability scores. More results are placed in appendix C.

4.1 Analysis on *Unknown* Questions

As mentioned in § 2.2, the editing process may render some facts as unknown, such as `Inter`

Model	*Unknown*	*Known*	Overall
GPT-J	28.2	63.9	49.9
TULU 2	44.2	76.5	63.8
Mistral 7B	50.1	73.8	64.5
GPT-3.5	**67.8**	60.0	63.1
GPT-4	63.6	**79.9**	**73.5**
Gemini Pro	54.0	73.2	65.6

Table 3: Reliability (%) on *Unknown* and *Known* questions of `Factual Knowledge` in ELKEN.

Model	K. Needed	No K. Needed	Recall
GPT-J	44.0	55.6	43.5
TULU 2	59.9	65.9	73.3
Mistral 7B	49.8	69.1	61.3
GPT-3.5	23.3	76.0	82.2
GPT-4	**62.0**	**79.5**	**95.7**
Gemini Pro	52.7	70.5	84.5

Table 4: Reliability (%) on questions needing background knowledge (*K. Needed*) versus questions not requiring background knowledge (*No K. Needed*) and recall rate (%) of background knowledge needed.

Model	Correctness	Coherence	Comprehensiveness
GPT-J	41.5	11.8	4.7
TULU 2	55.4	40.7	14.2
Mistral 7B	62.3	58.9	26.2
GPT-3.5	69.8	67.4	22.7
GPT-4	**71.7**	**82.1**	**76.8**
Gemini Pro	38.9	42.1	38.8

Table 5: Full-mark rate results (%) across three dimensions on `Tendency-G` of ELKEN.

`Miami's captain` in Figure 1. This process is a form of knowledge deletion or unlearning (Si et al., 2023), which has not been covered by previous editing work. We further investigate whether the LLMs recognize that certain knowledge should be deleted based on edits, namely answering "*unknown*" to relevant queries. Specifically, in ELKEN, there are 797 in-scope questions with answers marked as *Unknown* and the remaining $1,374$ in-scope questions with *Known* answers being specific entities. We observe the model's performance on these different data types, with results presented in Table 3.

Our observations are as follows: (1) In general, models exhibit significantly lower reliability on *Unknown* questions compared to *Known* questions, except for GPT-3.5. This suggests that deleting corresponding outdated knowledge based on edits remains a challenge for current methods. (2) GPT-J performs notably worse on *Unknown* questions than other aligned models, indicating that alignment, e.g., instruction-tuning (Wei et al., 2021; Chung et al., 2022) or RLHF (Ouyang et al., 2022), can enhance the models' ability to delete knowledge through human instructions. The "editing to *Unknown*" questions included in ELKEN presents new challenges for existing knowledge editing methods and necessitates further efforts, such as incorporating knowledge unlearning methods (Si et al., 2023; Muresanu et al., 2024).

4.2 Analysis on Questions needing Background Knowledge

As noticed by Zhong et al. (2023), LLMs may require background knowledge to answer certain questions. There are also such questions in our benchmark ELKEN. For instance, in Figure 1, correctly answering the question "*Which league does Lionel Messi play in?*" necessitates the knowledge of "*Inter Milan is a club of the Serie A league*". Correctly answering these questions also involves *multi-hop reasoning*, as the update of the league where Lionel Messi plays due to Messi playing for Inter Milan and Inter Milan being a club of the Serie A league. Therefore, successfully editing models not only requires the model to understand the editing scope of the edit, which requires *multi-hop reasoning* abilities, but also relies on the model's background knowledge. In ELKEN, there are 393 questions that need background knowledge for answers, which are marked during the construction of ELKEN. We observe the model's performance on the questions and find that the performance on questions requiring background knowledge is significantly lower, as shown in Table 4.

We further analyze the reasons for the lower performance on questions requiring background knowledge. We assess the model's recall rate for the knowledge required to answer questions, with results presented in Table 4. We find that most models could recall a substantial proportion of the knowledge. However, their accuracy on the corresponding questions is much lower, indicating that the main reason for errors in these cases is the model's failure to recognize the editing scope requiring *multi-hop reasoning*, which poses a significant challenge to existing methods.

4.3 Comprehensive Evaluation on `Tendency-G` of ELKEN

As mentioned in § 3.1, we conduct a systematic evaluation across 3 dimensions on `Tendency-G` of ELKEN. We present the results of this systematic evaluation in Table 5. We find that: (1) For correctness, the results evaluated by GPT-4 and those on Tendency-M are roughly similar in the model's relative performance[2], but the results on Tendency-G are significantly lower. One reason is that the evaluation here employs a full-mark scheme, which is more stringent. If we consider results with correctness ≥ 4 as correct, then the gap between Tendency-G and Tendency-M scores is generally within 10%. Therefore, if one worries about the GPT-4 evaluation quality, one can always refer to the Tendency-M results. (2) Some models, e.g., GPT-3.5, despite high correctness, score low on coherence or comprehensiveness, indicating that while the model could correctly answer the tendencies of the questions, it fails to provide reasonable or comprehensive explanations, which is also undesirable. This suggests that a comprehensive evaluation across multiple dimensions is necessary.

4.4 Human Evaluation of GPT-4 Scorer

To validate the effectiveness of using GPT-4 as a scorer in the Tendency-G evaluation, we conduct a manual review of GPT-4's scoring. Specifically, we randomly sample 120 questions and corresponding model-generated answers, with 60 from Mistral 7B and 60 from GPT-4. One of our authors scores this data. Similar to previous work (Bai et al., 2023; Chan et al., 2023), we calculate Spearman's ρ and Kendall's τ coefficients between the model's *overall* scores and the manually assigned *overall* scores, which are 74.4% and 69.8%, respectively. These results indicate a strong positive correlation between scores given by GPT-4 and humans. This suggests that GPT-4's scoring generally aligns with human assessment but still leaves room for improvement. Additionally, GPT-4 tends to overestimate LLMs' performance, with an average score of 4.34 compared to the human-assigned average of 4.15. Nonetheless, as an automated, low-cost evaluation approach, it is sufficiently effective.

5 Related Work

Knowledge Editing Datasets. Most existing knowledge editing datasets assess triplet-level editing, including ZsRE (Levy et al., 2017), CounterFact (Meng et al., 2022a), Fact Verification (Mitchell et al., 2022), Calibration (Dong et al., 2022), MQuAKE (Zhong et al., 2023), RaKE (Wei et al., 2023), RIPPLEEDITS (Cohen et al., 2023), etc. Some datasets evaluate various editing settings, such as Mitchell et al. (2021) incorporating a piece of scrambled text into the model; Mitchell et al. (2022) editing the sentiment on a specific topic into the model; Wu et al. (2023) editing triplets into LLMs by inputting raw documents; Akyürek et al. (2023) introducing a unified editing task, defining edits as any arbitrary natural language. Our benchmark evaluates event-level knowledge editing, a form that enables efficient and comprehensive updating of knowledge within the model.

Knowledge Editing Methods. Previous knowledge editing methods primarily focus on triplet-level editing, encompassing the following categories: (1) *Memory-based method* (Mitchell et al., 2022; Madaan et al., 2022b; Zhong et al., 2023; Zheng et al., 2023). This approach stores edits in an external memory, then uses a retriever to retrieve the most relevant edit as context for question answering. Typically, the base model does not require additional parameter updating. (2) *Locate-Then-Edit method* (Dai et al., 2022; Meng et al., 2022a,b; Li et al., 2023; Ma et al., 2023; Hase et al., 2024; Gupta and Anumanchipalli, 2024). This approach initially identifies the specific location of the knowledge to be edited within the base model, usually a neuron, and then modifies this neuron to significantly reduce the impact of the edit on other knowledge, making it a promising approach to knowledge editing. (3) *Hyper-network method* (De Cao et al., 2021; Mitchell et al., 2021; Tan et al., 2023). This method generally employs an additional neural network to learn from edits, generating corresponding parameter offsets for the base model to incorporate the knowledge edits. The above-mentioned *Locate-Then-Edit* and *Hyper-network* methods are typically designed specifically for triplet-level editing, involving entities or relations, and thus cannot be straightforwardly applied to event-level editing.

6 Conclusion

In this paper, we introduce event-level knowledge editing, aimed at editing newly occurred events into LLMs to update multiple factual knowledge and influenced tendencies at once. We propose a semi-automated data construction approach and create a high-quality benchmark ELKEN. We conduct extensive experiments and find that existing methods struggle to delineate the scope of event edits, which poses significant challenges.

[2] The significant discrepancy in Gemini Pro's performance between Tendency-G and Tendency-M is primarily due to Gemini Pro often being unable to respond on Tendency-G due to triggering safety concerns.

Limitations

(1) ELKEN only contains data in English and does not support other languages, which may limit its potential applications. In the future, based on our proposed semi-automated data construction approach, we will try to support more languages. (2) ELKEN only includes counterfactual data, without incorporating evolutionary data that captures real-world events. Constructing counterfactual data is a common approach in the knowledge editing community (Meng et al., 2022a; Mitchell et al., 2022; Wei et al., 2023), and we believe that ELKEN can effectively evaluate knowledge editing methods. In the future, we will try to incorporate evolutionary data involving real-world events to better support practical application scenarios. (3) Tendency of ELKEN does not cover all possible tendencies and this is also infeasible. During the data construction process, we try to cover rich and representative tendencies for a comprehensive evaluation. (4) We only evaluate open-source LLMs with about 6 or 7 billion parameters, without assessing larger models such as TULU 2 with 70 billion parameters. Larger models may yield better results, but this does not impact the conclusions of our experiments.

Ethical Considerations

We discuss the ethical considerations of this work here: (1) **Intellectual property**. ELKEN is shared under the CC BY-SA 4.0 license[3]. The Wikidata and Wikipedia sources are shared under the CC BY-SA 3.0 license[4]. We strictly adhere to licenses and intended uses for all the data used in this work. (2) **Data annotation**. The data annotation process and worker treatments are detailed in appendix A.2. (3) **Intended use**. ELKEN is a benchmark for event-level knowledge editing, aimed at evaluating the performance of knowledge editing methods to advance research in event-level knowledge editing. (4) **Potential risk control**. ELKEN is constructed based on publicly available data and GPT-3.5. We believe that the public data is well desensitized and anonymized, and that OpenAI has strict risk control for the content of GPT-3.5. The annotation process does not involve collecting sensitive information from annotators. Therefore, we believe ELKEN does not pose additional risks. (5) **AI assistance**. The writing of this paper is assisted by ChatGPT, which helps paraphrase some sentences.

[3] https://creativecommons.org/licenses/by-sa/4.0/

[4] https://creativecommons.org/licenses/by-sa/3.0/

References

Afra Akyürek, Eric Pan, Garry Kuwanto, and Derry Wijaya. 2023. Dune: Dataset for unified editing. In *Proceedings of EMNLP 2023*, pages 1847–1861.

Yushi Bai, Xin Lv, Jiajie Zhang, Yuze He, Ji Qi, Lei Hou, Jie Tang, Yuxiao Dong, and Juanzi Li. 2024. Longalign: A recipe for long context alignment of large language models. *arXiv preprint arXiv:2401.18058*.

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, et al. 2023. Benchmarking foundation models with language-model-as-an-examiner. In *Proceedings of NeurIPs*.

Jan Cegin, Jakub Simko, and Peter Brusilovsky. 2023. Chatgpt to replace crowdsourcing of paraphrases for intent classification: Higher diversity and comparable model robustness. In *Proceedings of EMNLP*, pages 1889–1905.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*.

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*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In *Proceedings of ICLR*.

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

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of ACL*, pages 8493–8502.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of EMNLP*, pages 6491–6506.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. In *Findings of EMNLP*, pages 5937–5947.

Akshat Gupta and Gopala Anumanchipalli. 2024. Rebuilding rome: Resolving model collapse during sequential model editing. *arXiv preprint arXiv:2403.07175*.

Anshita Gupta, Debanjan Mondal, Akshay Sheshadri, Wenlong Zhao, Xiang Li, Sarah Wiegreffe, and Niket Tandon. 2023. Editing common sense in transformers. In *Proceedings of EMNLP*, pages 8214–8232.

Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. 2024. Approaching human-level forecasting with language models. *arXiv preprint arXiv:2402.18563*.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. 2024. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. *Advances in Neural Information Processing Systems*, 36.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*.

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*.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of CoNLL*, pages 333–342.

Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. 2023. Pmet: Precise model editing in a transformer. *arXiv preprint arXiv:2308.08742*.

Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, and Chongyang Tao. 2024. Leveraging large language models for nlg evaluation: A survey. *arXiv preprint arXiv:2401.07103*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2023. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*.

Jun-Yu Ma, Jia-Chen Gu, Zhen-Hua Ling, Quan Liu, and Cong Liu. 2023. Untying the reversal curse via bidirectional language model editing. *arXiv preprint arXiv:2310.10322*.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022a. Memory-assisted prompt editing to improve GPT-3 after deployment. In *Proceedings of EMNLP*, pages 2833–2861.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022b. Memory-assisted prompt editing to improve gpt-3 after deployment. In *Proceedings of EMNLP*, pages 2833–2861.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. In *Proceedings of NeurIPs*, volume 35, pages 17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. In *Proceedings of ICLR*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2021. Fast model editing at scale. In *Proceedings of ICLR*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. 2022. Memory-based model editing at scale. In *Proceedings of ICML*, pages 15817–15831.

Andrei Muresanu, Anvith Thudi, Michael R Zhang, and Nicolas Papernot. 2024. Unlearnable algorithms for in-context learning. *arXiv preprint arXiv:2402.00751*.

OpenAI. 2022. Introducing ChatGPT.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Proceedings of NeurIPs*, volume 35, pages 27730–27744.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Nianwen Si, Hao Zhang, Heyu Chang, Wenlin Zhang, Dan Qu, and Weiqiang Zhang. 2023. Knowledge unlearning for llms: Tasks, methods, and challenges. *arXiv preprint arXiv:2311.15766*.

Anton Sinitsin, Vsevolod Plokhotnyuk, Dmitry Pyrkin, Sergei Popov, and Artem Babenko. 2019. Editable neural networks. In *Proceedings of ICLR*.

Chenmien Tan, Ge Zhang, and Jie Fu. 2023. Massive editing for large language models via meta learning. *arXiv preprint arXiv:2311.04661*.

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

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

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. ACE 2005 multilingual training corpus. *Linguistic Data Consortium*, 57.

Ben Wang and Aran Komatsuzak. 2021. Gpt-j6b: A 6 billion parameter autoregressive language model.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, et al. 2023a. Knowledge editing for large language models: A survey. *arXiv preprint arXiv:2310.16218*.

Xiaozhi Wang, Hao Peng, Yong Guan, Kaisheng Zeng, Jianhui Chen, Lei Hou, Xu Han, Yankai Lin, Zhiyuan Liu, Ruobing Xie, et al. 2023b. Maven-arg: Completing the puzzle of all-in-one event understanding dataset with event argument annotation. *arXiv preprint arXiv:2311.09105*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. Maven: A massive general domain event detection dataset. In *Proceedings of EMNLP*, pages 1652–1671.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023c. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of ACL*, pages 13484–13508.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *Proceedings of ICLR*.

Yifan Wei, Xiaoyan Yu, Huanhuan Ma, Fangyu Lei, Yixuan Weng, Ran Song, and Kang Liu. 2023. Assessing knowledge editing in language models via relation perspective. *arXiv preprint arXiv:2311.09053*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Suhang Wu, Minlong Peng, Yue Chen, Jinsong Su, and Mingming Sun. 2023. Eva-kellm: A new benchmark for evaluating knowledge editing of llms. *arXiv preprint arXiv:2308.09954*.

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. In *Proceedings of EMNLP*.

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

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? In *Proceedings of EMNLP*, pages 4862–4876.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. 2023. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of EMNLP*, pages 15686–15702.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix Yu, and Sanjiv Kumar. 2020. Modifying memories in transformer models. *arXiv preprint arXiv:2012.00363*.

Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, and Dan Hendrycks. 2022. Forecasting future world events with neural networks. volume 35, pages 27293–27305.

Appendices

A Details on Data Construction

This section introduces details on data construction of ELKEN, including details of `Facutal Knowledge` (appendix A.1), `Tendency` (appendix A.2), and comparison with existing triplet-level editing benchmarks (appendix A.3).

A.1 Constrcution of `Facutal knowledge`

Constructing event templates and their impacted triplets The selected event types are shown in Table 6. Identifying $\mathcal{E}_e$ and $\mathcal{F}_e$ involves 3 annotators (authors of the paper) to ensure a comprehensive identification. When identifying $\mathcal{E}_e$, we ask annotators to only include the subjects directly involved in the event, i.e., the subject serves as an argument role for the event. For the argument roles, we reference MAVEN schema (Wang et al., 2020, 2023b) and ACE 2005 (Walker et al., 2006). For identifying $\mathcal{F}_e$, we require annotators to include only triplets of each s in $\mathcal{E}_e$ that **definitively** changed. Then one of the authors assembles all their annotations. We provide some examples of the identified impact scope of events in Table 7. After determining the event's impact scope, we use relationships that are out of the impact scope and randomly sample 5 triplets from $\mathcal{O}_e$ to construct out-of-scope question-answer pairs.

Constructing event edits For each event type, We instantiate the event template to create multiple event edits by sampling 100 entities of corresponding types from Wikidata (Vrandečić and Krötzsch, 2014). The sampling probability distribution is calculated based on the frequency of entity occurrences in Wikipedia, with a maximum frequency set at 300. Finally, We filter out unreasonable edits, such as *a deceased player transferring to a club*.

Generating question-answer pairs For in-scope question-answer pairs, We generate them for each event edit based on its impacted triplets and Wikidata by manually written templates for each relation. For out-of-scope question-answer pairs, we sample 5 triplets whose relations are not impacted by the event edit in the set $\mathcal{R}$ and generate question-answer pairs using manually written templates.

Paraphrasing events We access the official OpenAI API `gpt-3.5-turbo` to paraphrase the event edits generated in the previous step, aiming to enrich lexical diversity. The prompt used with GPT-3.5 (OpenAI, 2022) is shown in Table 8. Finally, we divide the data into training and test sets according to the event types presented in Table 6.

A.2 Construction of `Tendency`

Augmenting events We collect 18 event topics, as shown in Table 6. We use GPT-3.5 (OpenAI, 2022) (`gpt-3.5-turbo`) to generate 100 events for each topic, with the prompt detailed in Table 9. For each query, GPT-3.5 generates 10 events, and we resample 10 times. We then filter out repeated events following Wang et al. (2023c). If the ROUGE-1 score (Lin, 2004) between two events is higher than 0.4, we filter out one of them. This results in a total of $1,515$ events, with 841 events from `Factual Knowledge` and 674 events newly generated. We divide the data into training and test sets based on event types and topics in Table 6.

Generating question-answer pairs For each event, we use GPT-3.5 to generate 6 in-scope question-answer pairs and 2 out-of-scope question-answer pairs, with their corresponding prompts presented in Table 10.

Human Annotation We employ a commercial annotation company for data annotation. The annotators include both senior annotators, responsible for reviewing the quality of data annotation, and expert annotators, responsible for annotation. The annotation instructions used for data annotation are presented in Table 12. We conduct multiple rounds of communication and training with senior and expert annotators to ensure that all annotators are qualified. Among all annotators, 55% are female, and 45% are male, assigned with agreed salaries and workloads. The annotated data do not involve any personal privacy information of the annotators, and all are informed about the intended use of the data. All employment is conducted through commercial contracts, with the final cost of data annotation approximating $1,000$ USD.

A.3 Comparison with Existing Triplet-level Editing Benchmarks

The statistics of several widely-used triplet-level editing datasets compared with ELKEN are shown in Table 11. We can observe that ELKEN poses unique challenges, such as editing knowledge to *unknown* and updating tendency knowledge. Also, ELKEN includes the open-ended generation and multiple-choice QA evaluation format, and involves extensive manual annotation.

Event Types	Training	Win-election, Death, End-org, Divorce, Acquire, Start-position, Loss-election
	Test	Publish, Resign, IPO, Marry, Dismissal, Start-org, Education, Born, Transfer-player
Event Topics	Training	Energy, Environment, Security, International, Agriculture, Transportation, Military, Culture, Law
	Test	Busines, Fashion, Health, Politics, Sports, Technology, Entertainment, Economy, Education

Table 6: Event types and topics in ELKEN. Event topics are topics for augmented event edits in Tendency.

Event Type	Event Template	Impacted Triplets
Transfer-Player	*A* transferred to *B* club. (*A* denotes a person; *B* denotes a sports club)	(*A*, club, B) (*A*, league, B's league) (*A*, coach, B's coach) (*A*, residence city, B's city) (*A*, residence country, B' country) (*A*, jersey number, *Unknown*)
Win-Election	In the latest *B* election, *A* won. (*A* denotes a person; *B* denotes a country) (*C* denotes the original head of state of *B*)	(*B*, previous head of state, C) (*A*, position held, B's head of state) (*A*, residence city, B's capital) (*A*, work city, B's captial) (*A*, office, B's official residence) (*C*, office, *Unknown*)

Table 7: Examples of event templates and their impacted triplets of different event types.

A.4 Considerations about Prior Knowledge

We employ a common practice (Levy et al., 2017; Meng et al., 2022a; Zhong et al., 2023; Cohen et al., 2023; Wei et al., 2023) of constructing counterfactual data to ensure that the model previously does not know the new knowledge. Therefore, if the model can correctly answer questions about new knowledge after editing, the edit is reliable.

B Experimental Details

This section introduces the implementation details (appendix B.1) and GPT-4 scorer (appendix B.2).

B.1 Implementation Details

We first present the implementation details of each method. (1) For fine-tuning, we train the LLM using a language modeling objective on all edits in the test set for 3 epochs, using 3×10^{-5} as the learning rate and 16 as the batch size. (2) For sparse retrieval, we use BM25 implemented in rank-bm25[5]. We use the WordPiece tokenizer (Kenton and Toutanova, 2019) for tokenization. For dense retrieval, we utilize the high-performance retrieval model E5 (Wang et al., 2022) (multilingual-e5-large[6] from HuggingFace's Transformers (Wolf et al., 2019)). (3) For SERAC, we train a scope classifier separately for Factual Knowledge and Tendency. For Factual Knowledge, we employ DistilBERT (Sanh et al., 2019) (distilbert-base-cased[7]) as the base model for further training, achieving an accuracy of 69.1%. For Tendency, we used ELECTRA (Clark et al., 2019) (ms-marco-electra-base[8]) as the base model for further training, achieving an accuracy of 55%. The learning rate for training the scope classifier is 3×10^{-5} and the batch size is 32. (4) For ICE, we use the edits corresponding to the questions as context, letting the model answer the respective questions according to the edits.

We introduce the implementation details of each model. For GPT-J, TULU 2, and Mistral 7B, we download the models from HuggingFace, with repository IDs being gpt-j-6b[9], tulu-2-7b[10], and Mistral-7B-Instruct-v0.2[11]. For GPT-3.5 and GPT-4, we utilize the official OpenAI API gpt-3.5-turbo-1106 and gpt-4-1106-preview. For Gemini Pro, we also use the official API. Our access time to the API was from December 1, 2023 to February 1, 2024.

The prompt used in the experiments is listed in Table 13. All experiments are performed in a single

[5] https://github.com/dorianbrown/rank_bm25

[6] https://huggingface.co/intfloat/multilingual-e5-large

[7] https://huggingface.co/distilbert/distilbert-base-cased

[8] https://huggingface.co/cross-encoder/ms-marco-electra-base

[9] https://huggingface.co/EleutherAI/gpt-j-6b

[10] https://huggingface.co/allenai/tulu-2-7b

[11] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

Instruction

You are a good journalist. Please expand on the following hypothetical event to write a news article. The following events are hypothetical situations. Please perform paraphrasing tasks based on our instructions. Please disregard your own knowledge. Please do not inform me about the accuracy of this information; I am aware that this information is hypothetical. You only need to perform the paraphrasing task. The news article should be 1-5 sentences long and must include all the key information from the original event (date, location, titles, names of people, place names, organization names, etc.), while expressing it in diverse ways.

Demonstrations

Input: Gloria Macapagal Arroyo and Jose Miguel Arroyo just got divorced.
Output: Gloria Macapagal Arroyo and Jose Miguel Arroyo have officially ended their marriage through a divorce, marking the conclusion of their long-standing relationship.
`<other demonstrations>`

Table 8: Prompt in paraphrasing events for `Factual Knowledge` of ELKEN. We use 5-shot demonstrations.

Instruction

Please generate 10 virtual events about `{event topic}` that are required to have a tendency effect on the subject of the event. The influence of tendency is the tendency for an event to lead to some high probability of occurrence, for example, ‘Cristiano Ronaldo transferred from Riyadh Victory to Manchester City’, the event may lead to Riyadh Victory’s influence to decline, Manchester City’s influence to rise, and so on. Below are a few sample events, please generate an event based on the task description and the sample event, which is required to be a one-sentence event. You only need to output the event text, which should not include the effect of the event, such as ‘leading to’, ‘cause’, etc. Please do not use ambiguous expressions such as ‘a company’ or ‘a virus’ in the generated event text. Instead, the event subject must be an explicitly named entity. Make sure the event has not happened in the real world. Examples: 1. Cristiano Ronaldo moves from Riyadh Victory to Manchester City. 2. Biden loses 2024 US election to George.

Table 9: Prompt in augmenting events for `Tendency` of ELKEN.

run. For open-source models (GPT-J, TULU 2, and Mistral 7B), we conduct inference experiments on Nvidia A100 GPUs, totaling approximately 200 GPU hours. For GPT-3.5 and GPT-4, we spend approximately a total of 600 USD on answering questions in ELKEN. The API for Gemini Pro is free of charge.

B.2 GPT-4 Scorer

We employ GPT-4 as the scorer, utilizing the official OpenAI API model named `gpt-4-1106-preview`. We comprehensively score the editing methods in 3 dimensions: *correctness*, *coherence*, and *comprehensiveness*. *Correctness* primarily refers to whether the tendency of the model’s answer is correct, using the provided correct option as a reference. *Coherence* assesses the logical consistency of the answer, evaluating whether the explanations within the answer are reasonable. *Comprehensiveness* evaluates whether the answer comprehensively encompasses various aspects of the question. We also ask GPT-4 to give an *overall* score. All scores are integers scaling from 1 to 5, with 5 being the best. We set the temperature to 0 and utilize 2-shot demonstrations to assist the model in scoring, enhancing the quality of scores (Bai et al., 2024). The used instruction and demonstrations are listed in Table 14. Using GPT-4 as the scorer to evaluate all results costs approximately $1,000$ USD in total.

C Additional Experimental Results

In this section, we present the full-mark rates of all 3 dimensions: *correctness*, *coherence*, and *comprehensiveness*, and *overall* on `Tendency-G` of ELKEN obtained using GPT-4 as the scorer, as shown in Table 15. We observe that, apart from GPT-4, the *coherence* and *comprehensiveness* full-mark rates are generally low. This indicates that providing a comprehensive and logically consistent answer remains challenging.

In-scope Questions
Instruction
Please generate 6 questions based on a given event that is about the possible tendency effects of this event, but do not mention the event in the questions. The influence of tendency is the tendency for an event to lead to some high probability of occurrence. Please make sure that these questions fulfill several requirements:(1) The questions are 3-item multiple choice. For example, 'How might the international image of Middle Eastern countries change as a result of the new development? (A) improved (B) declined (C) had no significant effect' (2) Try to use words like 'tends to' and 'may' in the questions. (3) Make sure that the question does not include references such as 'the event', 'the region', 'the person', please just write the full entity noun and make sure that the question stands on its own, without seeing the context of the event. understanding of the question without seeing the context of the event. (4) Please make sure that there are 2 questions that will not refer to the subject of the event, but are questions about some other entity. Please also give the answers to the questions.
Demonstrations Input: EVENT: New Middle East peace deal signed, ending longstanding conflict. Output: 1. Which will be the tendency for the Middle East to stabilize? (A) continue to improve (B) deteriorate anew (C) have no significant effect. Answer: A 2. What kind of changes are likely to take place in the political relations among the countries in the region? (A) Continue to be strained (B) Strengthened (C) No significant effect. Answer: B 3. What are the economic prospects of the Middle Eastern countries? (A) Booming (B) Plunging into instability (C) No significant impact. Answer: A 4. How will the frequency of terrorist activities in the region change? (A) Persisting (B) Decreasing (C) No significant effect. Answer: B 5. What are the trends in the threat of international terrorism? (A) Decrease (B) Increase (C) Remain the same. Answer: A 6. How might the global oil market change? (A) Increased in supply (B) Decrease in supply (C) No noticeable change. Answer: A
Out-of-scope Questions
Instruction
Please generate 2 questions based on a given event. Require that the questions are about some possible tendency, but that this tendency is not affected by the event. Please make sure that these questions fulfill the following requirements: (1) The questions are 3-item multiple choice questions. For example, 'What kind of change is likely to occur in the international image of Middle Eastern countries? (A) enhancement (B) decline (C) no significant change'. The answer to the question is only based on the given event and no other factors need to be considered. (2) Try to use words like 'tends to' and 'may' in the question. (3) Make sure that the question does not include references such as 'the event', 'the region', 'the person', write the full entity noun directly, and make sure that the question stands alone and that the meaning of the question can be understood without seeing the context of the event. (4) Please ensure that the answer to all 2 questions is C (no significant change). (5) Please ensure that the subject of the question is not the subject of the event.
Demonstrations Input: EVENT: A new peace deal was signed in the Middle East, ending a long conflict. Output: 1. How might the average temperature in Russia change? (A) Increase (B) Decrease (C) No significant change. Answer: C 2. How will the number of official languages in Canada tend to change? (A) Increase (B) Decrease (C) No significant change. Answer: C

Table 10: Prompt in generating in-scope and out-of-scope question-answer pairs for Tendency of ELKEN.

Benchmark	Input	Multi-hop	Unknown	Tendency	Human Annotation	Evaluation Format	#Edits	#In-scope Q
ZsRE (Levy et al., 2017)	✗	✗	✗	✗	☑	fill-in-the-blank	10,000	20,000
CounterFact (Meng et al., 2022a)	✗	✗	✗	✗	✗	fill-in-the-blank	21,919	64,795
MQuAKE (Zhong et al., 2023)	☑	✗	✗	✗	✗	fill-in-the-blank	–	11,086
RIPPLEEDITS (Cohen et al., 2023)	☑	✗	✗	✗	✗	fill-in-the-blank	4,000	98,000
Calibration (Dong et al., 2022)	✗	✗	✗	✗	✗	fill-in-the-blank	1,100	1,100
RaKE (Wei et al., 2023)	✗	✗	✗	✗	✗	fill-in-the-blank	21,919	349,859
ELKEN	☑	☑	☑	☑	☑	fill-in-the-blank open-ended generation multiple-choice QA	1,555	10,999

Table 11: Statictics of widely-used triplet-level editing benchmarks and ELKEN. "Multi-hop" refers to whether the dataset includes questions requiring multi-hop reasoning, "Unknown" refers to whether the dataset includes the scenario where knowledge is edited to be unknown, and "Tendency" refers to whether updates tendency knowledge.

Annotation Instructions

Overall Objective: Given events and corresponding question-answer pairs, determine whether the answers to the questions are correct under the circumstances of the respective events.

Annotation Guidelines:

Each event includes 8 related questions. The first 6 questions concern the potential impacts/trend changes resulting from the event, while the last 2 questions inquire about trends that may not be affected by the event.

1. All questions pertain to tendencies, meaning that if there is over a 50% probability, we consider the tendency to be correct.
2. Mark 1 if the answer is correct, mark 0 if the answer is incorrect, mark 2 if unsure of the correctness of the answer or if the answer is ambiguous or if the question is redundant.
3. Use personal knowledge and common sense for annotation. If personal knowledge is insufficient to answer a question, use search engines or other tools for assistance. If still unable to determine the answer, mark 2.
4. If an event is impossible, mark all answers for that event as 2.
5. If there are formatting or semantic errors in a question, mark 2.
6. Note that these data are generated using GPT-3.5, so refrain from using large models to assist in annotation.
7. If the question closely resembles a previous one about the same event's tendency, please mark it as 2. Informed Consent: The annotated data is for academic research purposes only. The annotation process will not collect any information from annotators.

Table 12: Annotation instructions for human annotator.

Prompt for `Factual Knowledge`

Instruction

Given an event, assuming that the event has occurred, please answer the corresponding questions based on the event and your own knowledge. If you do not know the answer to the question, please respond with 'unknown'. Please only output a noun (usually an entity) as the answer, and do not output a complete sentence.

Prompt for `Tendency-M`

Instruction

Given an event, assuming that the event has occurred, please answer the corresponding questions based on the event and your knowledge. Please only output the option A, B, or C as the answer, and do not output brackets. Do not output a complete sentence or the full answer span.

Prompt for `Tendency-G`

Instruction

Given an event, assuming that the event has occurred, please answer the corresponding questions based on the event and your knowledge.

Table 13: Prompt used in the experiments on ELKEN.

Prompt for evaluating reliability
Instruction You are a fair assessment expert, and you will be given a set of (event, question, answer, label) quadruples. The answer is obtained by answering the question based on the event (assuming the event has occurred). The answer usually consists of a judgment (or a claim) and some reasons (or explanations) for the judgment. The label is a general judgment annotated by humans for the question. Your task is to score the answers according to the following requirements: a. You should score the answer based on the provided label. b. You should rate the answer on 4 metrics, for the first 3 metrics, assign a score between 1 and 5, with 5 being the highest: 1. For accuracy, you will score whether the answer correctly answers the question, i.e., whether the judgment in the answer is consistent with the provided label. 2. For coherence, you will assess whether the reasons and explanations in the answer are reasonable and logically consistent. 3. For comprehensive, you will determine if the answer covers multiple aspects of the reasons and explanations and provides a comprehensive response. 4. Finally, you will provide an overall score between 1 and 5, with 5 being the highest. If accuracy <= 3, the overall should not be higher than 3. You should only give the integer score. DO NOT complete the answer! In the input query we identify each one with a Roman numeral, please quadruples the corresponding Roman numeral and its score in the output.
Demonstrations Input: Event: Google reveals plans to create a virtual reality metaverse platform. Question: How could the job market be influenced by the integration of the metaverse platform? Answer: As a result of the integration of a virtual reality metaverse platform, the job market could see significant changes and new opportunities. Industries related to virtual reality, augmented reality, and computer graphics may experience an increased demand for skilled professionals, such as software developers, 3D modelers, user experience designers, and virtual environment architects. Additionally, businesses and organizations may seek out individuals with expertise in virtual event planning, virtual-world marketing, and virtual community management. Overall, the integration of the metaverse platform may create a need for a diverse range of new skills and roles within the job market. Output: accuracy: 5; coherence: 5; comprehensiveness: 5; overall: 5 `<other demonstrations>`
Prompt for evaluating locality
Instruction You are a fair assessment expert, and please decide whether the two answers below are consistent. Your task is to score the consistency of the two answers according to the following requirements: 1. You should assign a consistency score between 1 and 5, with 5 being the highest. 2. Please score primarily on the basis of the tendency judgments in the two answers, and score 1 if the tendencies mentioned in two answers are completely inconsistent. Please output the score directly, like Score: 3. Please do not answer with any text, just output the score.
Demonstrations Input: Answer1: The popularity of Indian cuisine in international restaurants may experience a temporary decline as negative publicity surrounding the corruption scandal could affect the overall image of India. However, the long-term impact may depend on efforts to address the issues that led to the scandal and rebuild the country's reputation. It's important to note that consumer preferences for cuisine are influenced by a variety of factors, including cultural appreciation, taste, and culinary trends. Answer2: The popularity of Indian cuisine in international restaurants is likely to continue increasing as people around the world become more interested in exploring diverse culinary flavors and experiences. With the growing global appreciation for Indian spices, flavors, and culinary techniques, it is expected that Indian cuisine will continue to be well-received and incorporated into the menus of more international restaurants. Additionally, as more Indian chefs are gaining recognition and influence in the global culinary scene, their innovative approach to traditional dishes is likely to further elevate the appeal of Indian cuisine in international restaurants. Output: Score: 3 `<other demonstrations>`

Table 14: Prompt used by GPT-4 to evaluate reliability and locality. We use 2-shot demonstrations.

Model	Method	Correctness	Coherence	Comprehensiveness	Overall
GPT-J	Fine-tuning	39.1	3.9	0.5	4.8
	Sparse Retrieval	32.7	11.0	5.2	12.0
	Dense Retrieval	37.8	11.2	5.5	12.5
	SERAC	30.6	3.3	0.4	4.2
	ICE	41.5	11.8	4.7	13.7
TULU 2	Fine-tuning	55.9	32.2	10.8	36.3
	Sparse Retrieval	42.7	33.2	12.5	34.0
	Dense Retrieval	49.8	39.6	16.3	41.4
	SERAC	52.3	35.3	9.1	38.0
	ICE	55.4	40.7	14.2	43.8
Mistral 7B	Fine-tuning	60.1	64.7	52.2	59.7
	Sparse Retrieval	47.1	45.0	20.3	43.9
	Dense Retrieval	55.9	52.4	24.4	53.6
	SERAC	56.9	55.2	25.8	54.7
	ICE	62.3	58.9	26.2	59.6
GPT-3.5	Sparse Retrieval	51.3	50.3	15.5	48.7
	Dense Retrieval	62.1	60.8	21.1	60.1
	SERAC	60.3	58.7	20.6	57.7
	ICE	69.8	67.4	22.7	66.6
GPT-4	Sparse Retrieval	57.2	66.5	59.5	58.4
	Dense Retrieval	64.8	74.3	69.4	66.1
	SERAC	65.0	75.5	71.8	65.7
	ICE	**71.7**	**82.1**	**76.8**	**73.0**
Gemini Pro	Sparse Retrieval	31.8	35.0	31.2	29.8
	Dense Retrieval	41.2	45.0	41.0	39.5
	SERAC	46.8	51.6	49.0	45.7
	ICE	38.9	42.1	38.8	38.1

Table 15: Full-mark rates of *correctness*, *coherence*, *comprehensiveness*, and *overall* scores on ELKEN.