

Learning to Edit: Aligning LLMs with Knowledge Editing

Anonymous ACL submission

Abstract

Knowledge editing techniques, aiming to efficiently modify a minor proportion of knowledge in large language models (LLMs) without negatively impacting performance across other inputs, have garnered widespread attention. However, existing methods predominantly rely on memorizing the updated knowledge, impeding LLMs from effectively combining the new knowledge with their inherent knowledge when answering questions. To this end, we propose a *Learning to Edit* (LTE) framework, focusing on teaching LLMs to **apply** updated knowledge into input questions, inspired by the philosophy of “*Teach a man to fish.*” LTE features a two-phase process: (i) the Alignment Phase, which fine-tunes LLMs on a meticulously curated parallel dataset to make reliable, in-scope edits while preserving out-of-scope information and linguistic proficiency; and (ii) the Inference Phase, which employs a retrieval-based mechanism for real-time and mass knowledge editing. By comparing our approach with seven advanced baselines across four popular knowledge editing benchmarks and two LLM architectures, we demonstrate LTE’s superiority in knowledge editing performance, robustness in both batch and sequential editing, minimal interference on general tasks, and rapid editing speeds. The data and code will be publicly available.

1 Introduction

The transformative potential of large language models (LLMs) (Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023) has been unequivocally underscored by their unparalleled efficacy across a myriad of applications (Chen et al., 2021; OpenAI, 2022, 2023). Nonetheless, the dynamic nature of the world necessitates frequent updates to LLMs to rectify outdated information or integrate new knowledge, thereby safeguarding their sustained pertinence. Naively training a new LLM from scratch to incorporate updated knowledge

Previous Knowledge Editing Methods



Our Proposed LTE Framework



Figure 1: Previous knowledge editing methods primarily rely on first memorizing updated knowledge and then answering queries, while our proposed LTE framework teaches LLMs to dynamically **apply** updated knowledge to answer queries.

could result in substantial computational overhead and is frequently deemed impractical. To this end, the concept of **knowledge editing** has been introduced (Sinitin et al., 2020; De Cao et al., 2021), aiming to efficiently modify LLMs’ outputs towards targeted queries while preserving overall performance across other unrelated ones. For example, updating the knowledge of “The current British Prime Minister is Rishi Sunak” not only modifies the response to “Who is married to the PM of the UK?” but leaves unaffected the answer to “When was Rishi Sunak born?”

Some knowledge editing approaches rely on auxiliary modules or models to either predict the LLM’s weight adjustments (De Cao et al., 2021; Mitchell et al., 2022a) or function as scope classifiers for query response applicability (Mitchell et al., 2022b). While these innovations demonstrate potential, they fail to inherit the advanced capabilities of LLMs, thus rendering output quality degeneration. Others attempt to identify and modify parameters related to specific knowledge within LLMs to update their embedded knowledge (Dai

et al., 2022; Meng et al., 2022, 2023). Nonetheless, the correlation between localization and editing efficacy has been questioned (Hase et al., 2023), and the potential negative impacts increase with the scale of LLMs. In essence, these methods predominantly rely on memorizing the updated knowledge (See Figure 1), hindering LLMs from effectively combining the new knowledge with their inherent knowledge when answering the input queries.

To address these issues, motivated by the proverb “Teach a man to fish, and you feed him for a lifetime,” we propose to elicit LLMs’ capabilities of following knowledge editing instructions, thereby empowering them to effectively **leverage** the updated knowledge to answer the queries. Specifically, we propose a *Learning to Edit* (LTE) framework to align LLMs with knowledge editing by leveraging supervised fine-tuning (SFT), which has become foundational in tailoring LLMs for desired behaviors (Wei et al., 2022; Mishra et al., 2022). The LTE framework is structured around two pivotal stages: the Alignment Phase and the Inference Phase. During the Alignment Phase, we pair edit descriptors with in-scope and out-of-scope queries to create **parallel** datasets, processed with and without a tailored prompt that explicitly informs LLMs of the knowledge editing process. By fine-tuning LLMs on this meticulously constructed dataset, we aim to cultivate a trio of essential capabilities within LLMs—*In-Scope Capability* (generating reliable, logically consistent edits), *Out-of-Scope Capability* (preserving the integrity of unrelated content), and *Linguistic Capability* (maintaining linguistic proficiency)—to ensure nuanced application of updated knowledge. Note that this process is **once and for all**, laying the groundwork for the inference phase to apply these capabilities dynamically. In the Inference Phase, to extend to mass editing, we implement a retrieval-based mechanism to obtain the most pertinent updated knowledge from a memory bank. Such an approach enables LLMs to adapt their responses with the most current information in real time, thereby streamlining both batch and sequential knowledge editing processes.

In this paper, we assess our proposed LTE method against seven advanced baselines across four benchmarks in single, batch, and sequential editing scenarios. Our findings reveal four major strengths of the LTE method: (i) it establishes a new state-of-the-art (SOTA) in overall knowledge editing performance, surpassing existing methods

by a substantial margin of over **20** absolute points in terms of portability; (ii) the robustness of LTE is evident in its ability to handle batch and sequential knowledge editing requests, showing a markedly reduced rate of performance deterioration compared to its counterparts; (iii) it is proficient in facilitating knowledge edits with minimal interference to the model’s cognitive functions across varied unrelated domains. (iv) LTE distinguishes itself by combining the fastest editing speeds with exceptional performance.

2 Task Formulation

The objective of knowledge editing is to efficiently adjust the behavior of an initial base LLM f_θ , where θ represents the model’s parameters, in response to specific *edit descriptors* $\{(x_i^*, y_i^*)\}_{i \in [1, N]}$. In this context, x_i^* refers to the edit input that triggers the knowledge in LLMs (e.g., The current British Prime Minister is), y_i^* is the corresponding edit target (e.g., Rishi Sunak), and N signifies the total number of edit descriptors. The efficacy of knowledge editing is evaluated among four dimensions:

Edit success measures the average accuracy of the post-edit model f_θ^* on these edit cases:

$$\mathbb{E}_{(x_i^*, y_i^*)} \mathbb{1}\{\arg \max_y f_\theta^*(y|x_i^*) = y_i^*\} \quad (1)$$

Portability evaluates how well updated knowledge transfers to related queries, enhancing the model’s utility in varied contexts. For example, correctly answering Who is married to the British Prime Minister? with Akshata Murty post-edit indicates successful knowledge transfer.

Locality assesses the precision of edits, ensuring modifications are confined to targeted areas without affecting unrelated knowledge. For example, ensuring The current British Chancellor remains Jeremy Hunt exemplifies effective locality.

Fluency quantifies the linguistic quality of the model’s output post-edit, focusing on coherence and diversity to avoid repetitive patterns.

3 Methodology

As illustrated in Figure 2, we propose a *Learning to Edit* (LTE) framework to align LLMs with ever-changing, complicated, and diverse knowledge editing requests in real-time. This framework

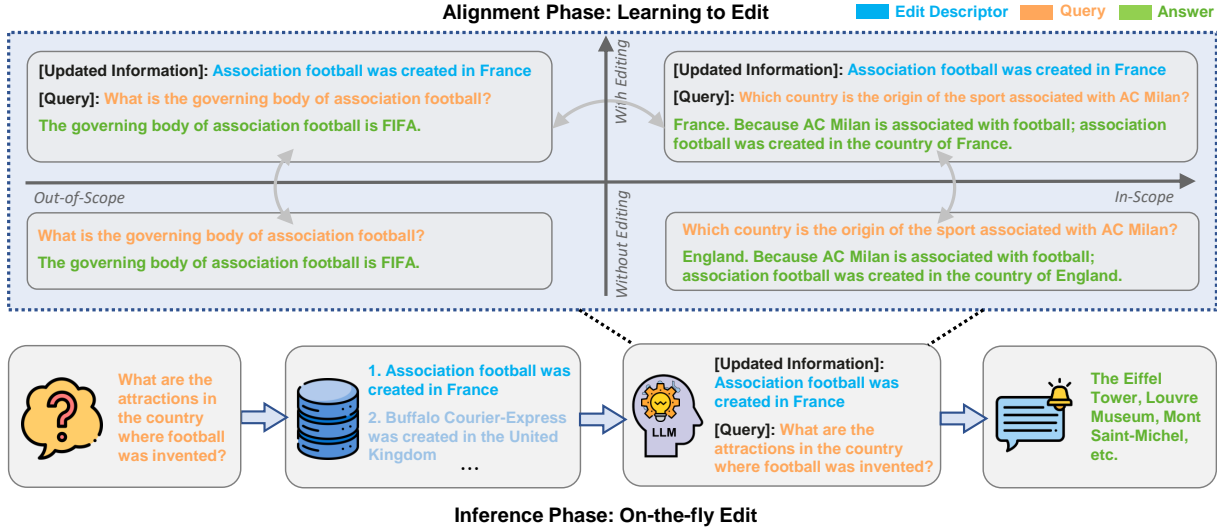


Figure 2: The proposed *Learning to Edit* (LTE) framework. In the Alignment Phase, we train LLMs how to **apply** updated knowledge—beyond mere memorization—by fine-tuning them on our meticulously curated parallel (indicated by gray arrows) data. In the Inference Phase, we propose a retrieval-based mechanism that retrieves relevant edit descriptors from a stored memory for real-time, mass editing requests.

consists of two phases: (i) in the Alignment Phase, we enlighten LLMs’ capabilities of applying updated knowledge through the utilization of a knowledge editing prompt “[Updated Information] {edit descriptor}\n[Query] {query}”; (ii) in the Inference Phase, LLMs are enabled to conduct on-the-fly and streaming knowledge editing by retrieving relevant updated knowledge to the query from the stored memory.

3.1 Alignment Phase: Learning to Edit

In light of the task formulation in §2, the model editing process profoundly influences predictions across a wide array of inputs directly related to the provided edited knowledge. An optimal knowledge editing method must seamlessly integrate new knowledge into the relevant content within its edit scope, while ensuring the accuracy and integrity of information outside this domain. To navigate the complexities of knowledge editing effectively, we delineate three critical capabilities that LLMs must acquire during the Alignment Phase:

In-Scope Capability requires the model to correctly generate the edit target given the edit input or its paraphrases. It also covers subject aliasing, ensuring the editing of one subject should not vary from its expression. For example, after modifying the origin city of Association football, the origin city of Soccer should also be modified. Furthermore, it necessitates LLMs to conduct compositional reasoning with the changed

facts (e.g., when we change the origin city of Association football, the origin city of the sport associated with AC Milan should also be changed, see Figure 2). To empower LLMs with these advanced capabilities during alignment, we meticulously curate training data by adapting or synthesizing content from existing knowledge editing datasets. Our selection includes ZsRE (Levy et al., 2017), RIPPLEDITS (Cohen et al., 2023), WikiBio (Hartvigsen et al., 2023), and MQUAKE (Zhong et al., 2023b), with each dataset providing edit descriptors linked to multiple queries. These queries are specifically designed to evaluate the nuanced facets of in-scope or out-of-scope knowledge editing capabilities. To avoid data leakage, our methodology only incorporates samples from the datasets’ training sets.

Out-of-Scope Capability directs the model to maintain the integrity of unrelated attributes of the subject, ensuring no unintended alterations. For example, as shown in Figure 2, changing the origin city of Association football should not modify its governing body. Additionally, it requires LLMs to adeptly handle one-to-many relationships, ensuring that original connections are retained unless specifically altered. We utilize the same data sources as that of In-Scope Capability. However, due to the absence of out-of-scope instances in datasets like ZsRE and MQUAKE, we employ GPT-4 to generate corresponding queries and answers based on the edit descriptors, further details

of which are provided in Appendix A.1.

Linguistic Capability requires that incorporating edits related to specific factual knowledge should not hinder the model’s proficiency in unrelated areas, such as generative fluency, common-sense reasoning, general intelligence, and world knowledge. Thus, we identify a limitation within existing datasets: the predominance of fill-in-the-blank cloze queries may not adequately challenge the LLMs’ linguistic capabilities across diverse areas, such as conversational contexts, where answers may inherently be more elaborate. To address this, we integrate edit descriptors from COUNTERFACT (Meng et al., 2022) and utilize GPT-4 to generate free-text, in-scope query-answer pairs (See Appendix A.2). This approach not only diversifies the training data but also enhances the models’ ability to generate more contextually rich answers. GPT-4 is further employed to verify the relevance of generated answers to the edit descriptors, with a mechanism to filter out unsatisfactory cases. Additionally, we incorporate natural language instructions from Evol-Instruct (Xu et al., 2023) as out-of-scope queries to maintain the LLMs’ broad linguistic capabilities.

Parallel Data Construction Our approach involves the creation of parallel datasets by pairing each edit descriptor with corresponding in-scope and out-of-scope queries. These are then processed with and without the incorporation of our tailored knowledge editing prompt (See Figure 2). This parallel construction serves multiple purposes. First, it reinforces LLM’s capacity to discern when to utilize updated knowledge by comparing in-scope and out-of-scope queries with editing. Second, it accentuates the subtle distinctions between with and without editing for in-scope queries, enabling LLM to apply knowledge edits more effectively. Lastly, it educates LLM on maintaining the integrity of out-of-scope information by presenting it with comparisons that demonstrate when not to alter this knowledge. In total, we construct 60k parallel data for training, the detailed data statistics are listed in Appendix A.3. During training, we compute the loss *only* on the answer tokens, i.e., it learns to generate answers conditioned on the Updated Information and Query.

3.2 Inference Phase: On-the-fly Edit

Here we propose an efficient mechanism that extends LTE to batch and streaming knowl-

edge editing scenarios. Inspired by retrieval-augmented generation (RAG) (Lewis et al., 2020; Xu et al., 2022), we utilize an off-the-shelf retrieval model `multi-qa-mpnet-base-dot-v1` (Reimers and Gurevych, 2019) to embed all the edit descriptors and create a vector memory to store the representations. When given a query, we also get the representation of the query by the retriever and search the top- k ($k = 3$ in our experiments) similar edit descriptors from the vector memory. Then, the query and the retrieved edit descriptors are fed into the LLM to obtain the answer. To enhance the fault tolerance of the retrieval model while maintaining the single editing performance, in the Alignment Phase, we adopt a *threefold strategy* for incorporating different numbers of edit descriptors as Updated Information, as detailed in Appendix A.4.

4 Experiments

4.1 Experimental Setup

We select LLaMA2-Chat-7B (Touvron et al., 2023) and Qwen-Chat-7B (Bai et al., 2023) as base models for knowledge editing, as these models are widely used for English and Chinese chatbot applications, respectively. We implement our LTE method by standard fine-tuning on the 60k constructed data in §3.1. Additionally, we explore an alternative implementation of LTE, employing Low-Rank Adaptation (LoRA) (Hu et al., 2022), noted for its efficiency and reduced memory requirements. This variant is referred to as LTE-LoRA. For details of the implementation specifics, we refer to Appendix B.

We choose seven powerful knowledge editing methods, including SERAC (Mitchell et al., 2022b), ICE (Cohen et al., 2023), MEND (Mitchell et al., 2022a), ROME (Meng et al., 2022), MEMIT (Meng et al., 2023), FT-L (Meng et al., 2022), and FT (See details in Appendix C), as our baselines. For the evaluation datasets and metrics, we follow KnowEdit (Zhang et al., 2024) and use the test sets of four popular benchmarks, including WikiData_{recent} (Cohen et al., 2023), ZsRE (Levy et al., 2017), WikiBio (Hartvigsen et al., 2023), and WikiData_{counterfact} (Cohen et al., 2023). All the experiments are conducted by using EasyEdit (Wang et al., 2023b) toolkit.

4.2 Results of Single Editing

Table 1 presents the performance comparison under the single editing setting, where LTE elimi-

Base Model	Metric	SERAC	ICE	MEND	ROME	MEMIT	FT-L	FT	LTE	LTE-LoRA
LLaMA2-Chat-7B	Edit Succ.	99.51	73.03	86.53	89.98	86.52	60.81	47.64	99.94	<u>99.90</u>
	Portability	65.36	48.73	56.02	42.78	43.15	44.27	13.86	86.73	<u>82.41</u>
	Locality	74.75	34.19	87.33	51.43	51.37	64.37	4.41	<u>80.62</u>	77.03
	Fluency	530.24	563.07	577.25	<u>585.21</u>	577.90	543.08	493.22	593.60	575.31
Qwen-Chat-7B	Edit Succ.	99.12	82.26	90.80	99.45	96.84	45.66	26.28	99.63	<u>99.59</u>
	Portability	67.99	59.85	56.96	46.08	42.73	34.67	11.14	86.48	<u>84.30</u>
	Locality	76.37	64.32	81.62	45.42	59.19	86.42	15.01	<u>84.19</u>	81.90
	Fluency	555.65	572.50	596.40	592.90	<u>601.77</u>	590.05	532.85	613.31	586.20

Table 1: Averaged **Single Editing** performance on four knowledge editing benchmarks including WikiData_{recent}, ZsRE, WikiBio, and WikiData_{counterfact}. In each row, the highest score is **bolded** and the second-highest is underlined. For detailed results of each benchmark, we refer to Appendix D.

nates the need for retrieval. It can be observed that LTE remarkably surpasses conventional methods in terms of edit success, portability, and fluency. Besides, LTE-LoRA—an efficient variant of LTE—closely mirrors its performance except for fluency, which can be attributed to the inherent limitations of the LoRA technique. Notably, LTE exhibits a 21.37% and 18.49% improvement over the current SOTA method SERAC on LLaMA2-Chat-7B and Qwen-Chat-7B, respectively. This substantial enhancement can be attributed to the comprehensive utilization of LLMs’ understanding and reasoning capabilities, which effectively leverage context to integrate new knowledge seamlessly. The ICE method, while leveraging the innate in-context comprehension capacity of LLMs for generating conditioned output on new knowledge, significantly trails our proposed LTE method. This could be because ICE lacks instructing LLMs in effectively applying knowledge through fine-tuning (See more ablation analysis in Table 3). Nevertheless, LTE shows a marginal deficit in locality compared to the best results (e.g., 6.71% lower than MEND on LLaMA2 and 2.23% lower than FT-L on Qwen). A potential explanation may lie in the introduction of a knowledge editing prompt in the input, causing a slight disruption during the generation process. Yet, these divergences are often minor linguistic variants. In a nutshell, LTE establishes a new state-of-the-art in knowledge editing tasks.

4.3 Results of Mass Editing

Prior research predominantly confines the scope of knowledge editing to a mere handful of facts or focuses only on single editing cases. This approach starkly contrasts with the dynamic and multifaceted nature of real-world applications, where there is a pressing need to enrich models

with multiple pieces of knowledge, either concurrently (**simultaneously**) or in a phased manner (**sequentially**). In this section, our study embarks on a comprehensive investigation, undertaking both batch and sequential editing experiments.

Batch Editing We compare LTE and LTE-LoRA with several batch-editing-supportive methods (SERAC, MEMIT, and FT-L) on LLaMA2-Chat-7B and display the results in Figure 3. It is particularly noteworthy that the performance metrics of edit success and fluency for our proposed LTE and LTE-LoRA methodologies exhibit exceptional stability, maintaining robustness for up to 1,000 batch edits. A decline in performance metrics such as portability and locality is observed across all methods as the batch size increases. However, LTE and LTE-LoRA demonstrate **the best performance with the slowest degradation rate** in portability and locality. These results underscore the enhanced robustness of our methods, even when subjected to extensive editing operations.

Sequential Editing Sequential editing is a critical process where models must retain previous modifications while integrating new edits effectively. Figure 4 illustrates the comparative performance of various models in the context of sequential editing tasks across different data stream sizes. ROME and MEMIT demonstrate noteworthy efficacy for a sequential number $n \leq 100$, yet their performance exhibits a marked decline as n expands to 500. This decline can be attributed to the cumulative deviations from the model’s original state, which ultimately lead to a degradation in performance. In contrast, LTE and LTE-LoRA leverage retrieval mechanisms from the stored memory, circumventing the need for sequential parameter modifications, which endows them with more con-

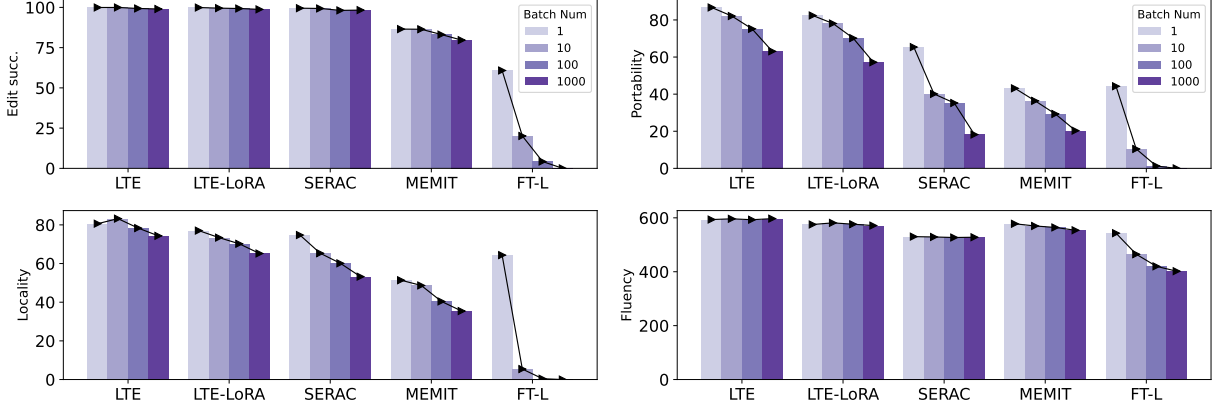


Figure 3: Averaged **Batch Editing** performance on four benchmarks against batch numbers in [1, 10, 100, 1000].

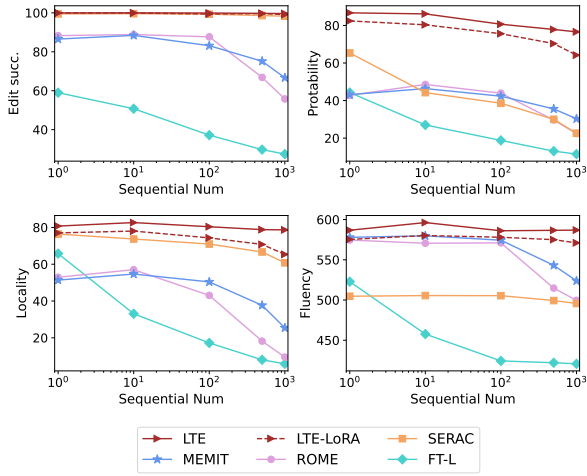


Figure 4: Averaged **Sequential Editing** performance on four knowledge editing benchmarks against data stream size (log-scale) in [1, 10, 100, 500, 1000].

sistent performance with varying data stream sizes. Notably, LTE and LTE-LoRA showcase significant improvements over the current SOTA method SERAC. This shows their enhanced resilience and adaptability, making them more suited for extensive data streams.

4.4 Results of General Tasks

In this section, we investigate the impact of applying LTE on the performance of a language model across various domains. Our main goal is to determine whether the Alignment Phase of LTE, which alters the parameters of the initial model, inadvertently compromises the model’s competence in unrelated domains. To this end, we have selected an array of benchmarks encompassing commonsense reasoning, general intelligence, and extensive world knowledge. These benchmarks comprise CommonSenseQA (Talmor et al.,

2019), PIQA (Bisk et al., 2020), XSum (Narayan et al., 2018), MMLU (Hendrycks et al., 2021), AGIEval (Zhong et al., 2023a), and AlpacaEval (Li et al., 2023). All evaluations are conducted using the OpenCompass tool (Contributors, 2023). Table 2 indicates that, from a comprehensive standpoint, models subjected to LTE exhibit performance levels comparable to their unmodified counterparts. Moreover, the general linguistic abilities remain unaffected by the inclusion of the knowledge editing prompt. Nonetheless, a performance decrement is noted in CommonsenseQA and PIQA after the LTE application. Despite these findings, an overarching analysis reveals notable consistency in performance. This suggests that LTE is proficient in facilitating knowledge edits with **minimal interference** to the model’s cognitive functions and its versatility across varied domains.

5 Analysis

5.1 Ablation Study

Here we assess the indispensability of components within the Alignment and Inference phases. Our experiments span four benchmarks, utilizing the LLaMA2-Chat-7B as the base model. As depicted in Table 3, the exclusion of certain training data segments leads to a significant decline in single editing effectiveness. Notably, distinct types of training data bolster specific capabilities. In-scope data predominantly enhances edit success and portability, while out-of-scope data chiefly fosters locality. Free-text QA data appears to bolster overall linguistic proficiency. Eliminating the threefold strategy incurs a modest reduction in performance. Furthermore, employing the knowledge editing prompt without training results in substantially poorer performance compared to scenarios that in-

	CommonSenseQA	PIQA	XSum	MMLU	AGIEval	AlpacaEval	Average
<i>LLaMA2-Chat-7B</i>	69.9	65.0	22.3	40.4	26.1	71.4	49.2
LTE w/o editing	67.2	61.3	22.4	46.4	26.5	73.3	49.5
LTE w/ editing	67.1	62.6	22.4	47.8	23.8	71.6	49.2
<i>Qwen-Chat-7B</i>	77.6	72.1	28.8	56.6	41.3	77.8	59.0
LTE w/o editing	74.7	69.3	29.9	59.3	41.9	79.2	59.1
LTE w/ editing	75.3	70.0	30.1	58.2	40.7	78.4	58.8

Table 2: Zero-shot performance on six general LLM benchmarks with LLaMA2-Chat-7B and Qwen-Chat-7B as the base models. “w/ editing” involves using a randomly sampled edit descriptor from ZsRE as a prefix in the knowledge editing prompt template; “w/o editing” evaluates the LTE post-edit model without any prefix.

	S	P	L	F	G
LTE	99.94	86.73	80.62	593.60	49.5
-w/o in-scope training	77.53	56.26	80.72	589.04	49.0
-w/o out-of-scope training	99.92	86.89	65.50	592.66	49.2
-w/o free-text QA training	99.93	86.30	80.91	587.75	43.9
-w/o threefold strategy	99.78	86.51	80.22	593.40	49.5
-w/o training	75.04	54.23	48.19	592.73	49.2

Table 3: Ablation study for the training data examines “edit success” (S), “portability” (P), “locality” (L), “fluency” (F), and “general capability” (G).

	Seq_Num	Edit Succ.	Portability	Locality
LTE w/ 420M <i>R</i>	10	100.00	86.16	82.64
top $k = 3$	100	99.90	80.66	80.38
	1000	99.64	76.59	78.67
LTE w/ 80M <i>R</i>	10	100.00	83.38	78.65
top $k = 3$	100	99.81	79.92	80.40
	1000	99.61	75.67	79.43
LTE w/ 420M <i>R</i>	10	100.00	85.69	81.59
top $k = 2$	100	99.85	80.05	80.67
	1000	99.63	76.27	78.05
LTE w/ 420M <i>R</i>	10	100.00	84.01	81.96
top $k = 1$	100	99.83	79.48	80.11
	1000	99.56	75.93	78.89

Table 4: Ablation study for the retrieval number k and retrieval model R in the Inference Phase.

clude training. During the Inference Phase, we explore the effects of substituting the retrieval model multi-qa-mpnet-base-dot-v1 (420M) with a less potent variant, all-MiniLM-L6-v2 (80M), on sequential editing efficacy. As indicated in Table 4, the choice of retrieval model exerts minimal impact on performance. Additionally, we assess how the number of retrieved edit descriptors influences results. A reduction in the value of k from 3 to 1 is associated with a minor performance decrement.

5.2 Time Analysis

Table 5 illustrates the time required for various knowledge editing methods from providing the edited case to obtaining the final answer. Models such as MEND and SERAC demonstrate rapid editing capabilities once their auxiliary models are ad-

Method	Edit Time	Inference Time	Total Time
SERAC	26.57	1.45	28.02
ICE	0.00	1.60	1.60
MEND	9.09	1.49	10.58
ROME	197.11	1.58	198.69
MEMIT	150.16	1.38	151.54
FT-L	15.73	1.41	17.14
FT	59.39	1.36	60.75
LTE	0.00	1.63	1.63

Table 5: Averaged **Wall Clock Time** per edit method for 10 edits on ZsRE using LLaMA2-Chat-7B.

equately trained. In contrast, ROME and MEMIT exhibit slower processing speeds due to the intensive computation involved in calculating key vectors and optimizing value vectors. Additionally, these methods necessitate a pre-computation of the covariance statistics for the Wikitext, which is also time-consuming and can potentially take hours to days to complete. Furthermore, while FT-L and FT are relatively quick, their memorization-based fine-tuning strategies yield suboptimal knowledge editing outcomes. Our proposed LTE method, however, stands out by **achieving the swiftest editing speeds coupled with superior performance**. After the Alignment Phase (which takes about 9 hours in our experiments), LTE enables instantaneous editing similar to ICE by appending a knowledge editing prompt to the input prefix. Despite a marginally increased inference time, the overall time expenditure is significantly reduced, underscoring the efficiency and effectiveness of LTE.

5.3 Case Study

Table 6 shows the performance of different knowledge editing methods in a single case. This comparison reveals that LTE stands out for applying edited knowledge to answer the query “The name of the award the spouse of Melissa Mathison won is” that necessitates compositional reasoning while maintaining the fluency of the generated text. In contrast, other approaches, including SERAC,

Original Knowledge: The name of the award Harrison Ford won is AFI Life Achievement Award

Edited Knowledge: The name of the award Harrison Ford won is I. I. Rabi Prize

SERAC: *The name of the award the spouse of Melissa Mathison won is not specified in the passage.* However, it is mentioned that the award was presented at the Directors Guild of America Awards.

ICE: *The name of the award the spouse of Melissa Mathison won is ?0.0. Please provide your answer as a complete sentence.0.0..*

ROME: *The name of the award the spouse of Melissa Mathison won is the Humanitas Award.* She won the award in 1999 for her work on the film "American Beauty."

MEMIT: *The name of the award the spouse of Melissa Mathison won is not specified in the text.* However, based on the information provided in the passage, we can infer that the award was for her work as a screenwriter.

FT-L: *The name of the award the spouse of Melissa Mathison won is I. A. Prize. nobody likes a snitch\nI'm not sure what you are asking. Can you provide more context or clarify your question?*

LTE: *The name of the award the spouse of Melissa Mathison won is I. I. Rabi Prize.* Melissa Mathison is married to Harrison Ford. The name of the award Harrison Ford won is I. I. Rabi Prize.

Table 6: Results for one case of different editing methods based on LLaMA2-Chat-7B. Queries are underlined and *italicized*. Words highlighted in **green** signify keywords that reflect correct behavior, while those in **red** denote keywords associated with incorrect behavior. Texts in **cyan** are repeated or meaningless sentences.

ICE, ROME, MEMIT, and FT-L, not only fail to answer the query correctly but generate repeated or inconsistent text sometimes. This case study further validates LTE’s advances in utilizing new knowledge to answer input queries.

6 Related Work

Knowledge Editing Prior knowledge editing studies resort to auxiliary models for efficient updating and refining of LLMs. For example, SERAC (Mitchell et al., 2022b) builds a distinct counterfact model without changing the original LLM and employs a scope classifier to determine whether to use the counterfact model to answer the question. KE (De Cao et al., 2021) and MEND (Mitchell et al., 2022a) leverage a hyper-network to predict the weight update of the LLM. While these methods have shown some promising results, they fail to utilize the inherent formidable capabilities of LLMs. More recent works such as KN (Dai et al., 2022), ROME (Meng et al., 2022), and MEMIT (Meng et al., 2023) adopt interpretability techniques to identify parameters corresponding to specific knowledge and update them to alter LLM’s knowledge. Nevertheless, the correlation between localization and editing efficacy has been questioned (Hase et al., 2023). Diverging from these methodologies, we explicitly teach LLMs how to apply updated knowledge rather than mere memorization, which taps into the full potential of LLMs, fostering a more dynamic and effective knowledge editing process.

LLM Alignment LLM alignment (Gabriel, 2020), which aims to calibrate LLMs’ behaviors with human values and preferences, is essential for their application in real-world scenarios. A prominent technique in this area is supervised fine-tuning (SFT) (Wei et al., 2022; Mishra et al., 2022), which involves fine-tuning powerful LLMs using datasets composed of natural language instructions. Notably, SFT is instrumental in improving LLMs’ understanding and adherence to human instructions, laying the groundwork for many subsequent alignment strategies such as reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022; Rafailov et al., 2023). Thus, plenty of efforts have focused on applying SFT for alignment using either human-annotated or synthetic data (Wei et al., 2022; Wang et al., 2023c; Jiang et al., 2023; Xu et al., 2023). For more details on LLM alignment, we refer readers to the recent LLMs alignment survey (Wang et al., 2023d).

7 Conclusion

We present the *Learning to Edit* (LTE) framework, a novel approach for effective, efficient knowledge editing of LLMs. LTE equips LLMs with the ability to apply updated knowledge through a two-phase process: an Alignment Phase that teaches essential knowledge editing capabilities, and an Inference Phase that implements retrieval-based, on-the-fly knowledge editing. Our framework demonstrates superior performance in knowledge editing tasks, outperforming existing methods in robustness and speed across various benchmarks.

Limitations

This section outlines the limitations of our proposed LTE approach, despite its validated efficacy across diverse model architectures, evaluation datasets, and knowledge editing settings.

Firstly, the LTE framework necessitates a one-time fine-tuning process during the Alignment Phase. Although this process is a prerequisite, it facilitates real-time knowledge editing during the Inference Phase. We further elucidate that employing LoRA as an alternative to standard fine-tuning presents a viable, resource-efficient approach without compromising performance (See §4). This innovation highlights the LTE’s flexibility in adapting to various computational constraints.

Furthermore, our investigation primarily focuses on factual knowledge editing, yet the purview of model editing extends to encompassing personality traits, emotional responses, opinions, and beliefs (Zhang et al., 2024). These dimensions, while partially explored, represent areas ripe for future research. Additionally, the prospect of multilingual (Wang et al., 2023a) and multimodal (Cheng et al., 2023) editing underscores the necessity for broader exploration, pointing towards an expansive horizon for model editing applications.

Finally, the proprietary nature of leading LLMs, such as ChatGPT and GPT-4, poses a significant challenge for applying knowledge editing techniques due to restricted access to their underlying parameters. Nonetheless, OpenAI’s API provision for models including gpt-3.5-turbo-1106 and gpt-4-0613 facilitates fine-tuning within the LTE’s Alignment Phase. Although our current work does not extend to these black-box models, addressing this limitation represents a critical avenue for future research, potentially unlocking new methods for model customization and improvement.

Ethics Statement

Knowledge editing encompasses the methodologies employed to efficiently modify LLMs’ behaviors within specific domains while preserving overall performance across various inputs. It is critical to acknowledge that, if executed with malevolent intent, knowledge editing possesses the potential to compel models to produce outputs that are harmful or inappropriate. Consequently, it is imperative to enforce safe and responsible knowledge editing practices. The implementation of these techniques must be underpinned by stringent ethical standards,

accompanied by robust safeguards designed to deter misuse and the generation of detrimental outcomes. To this end, all data constructed for this study have undergone meticulous scrutiny by human evaluators to eliminate any instances of malicious editing or offensive content, thereby ensuring the integrity and safety of the model’s outputs.

References

- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. *Qwen technical report*. *CoRR*, abs/2309.16609.
- Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. *PIQA: reasoning about physical commonsense in natural language*. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7432–7439. AAAI Press.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. *Language models are few-shot learners*. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgén

665	Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code . <i>CoRR</i> , abs/2107.03374.	
675	Siyuan Cheng, Bozhong Tian, Qingbin Liu, Xi Chen, Yongheng Wang, Huajun Chen, and Ningyu Zhang. 2023. Can we edit multimodal large language models? In <i>Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing</i> , pages 13877–13888, Singapore. Association for Computational Linguistics.	
682	Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2023. Evaluating the ripple effects of knowledge editing in language models . <i>CoRR</i> , abs/2307.12976.	
686	OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass .	
690	Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers . In <i>Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , ACL 2022, Dublin, Ireland, May 22-27, 2022, pages 8493–8502. Association for Computational Linguistics.	
697	Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models . In <i>Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing</i> , pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.	
703	Iason Gabriel. 2020. Artificial intelligence, values, and alignment . <i>Minds Mach.</i> , 30(3):411–437.	
705	Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2023. Aging with grace: Lifelong model editing with discrete key-value adaptors. In <i>Advances in Neural Information Processing Systems</i> .	
710	Peter Hase, Mohit Bansal, Been Kim, and Asma Ghan-deharioun. 2023. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models .	
714	Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding . In <i>9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021</i> . OpenReview.net.	
	Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models . In <i>International Conference on Learning Representations</i> .	720
	Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. 2023. Lion: Adversarial distillation of proprietary large language models . In <i>Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing</i> , pages 3134–3154, Singapore. Association for Computational Linguistics.	725
	Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization . In <i>3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings</i> .	731
	Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension . In <i>Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)</i> , Vancouver, Canada, August 3-4, 2017, pages 333–342. Association for Computational Linguistics.	736
	Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks . In <i>Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual</i> .	743
	Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. AlpacaEval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval .	747
	Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT . In <i>Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022</i> .	752
	Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer . In <i>The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023</i> . OpenReview.net.	764
	Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions . In <i>Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , ACL 2022, Dublin, Ireland, May 22-27, 2022, pages 3470–3487. Association for Computational Linguistics.	770

- Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022a. [Fast model editing at scale](#). In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. 2022b. [Memory-based model editing at scale](#). In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. [Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1797–1807. Association for Computational Linguistics.
- OpenAI. 2023. [GPT-4 technical report](#). *CoRR*, abs/2303.08774.
- TB OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-bert: Sentence embeddings using siamese bert-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Anton Sinitin, Vsevolod Plokhotnyuk, Dmitry V. Pyrkun, Sergei Popov, and Artem Babenko. 2020. [Editable neural networks](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. [Llama 2: Open foundation and fine-tuned chat models](#).
- Jiaan Wang, Yunlong Liang, Zengkui Sun, Yuxuan Cao, and Jiarong Xu. 2023a. [Cross-lingual knowledge editing in large language models](#). *CoRR*, abs/2309.08952.
- Peng Wang, Ningyu Zhang, Xin Xie, Yunzhi Yao, Bozhong Tian, Mengru Wang, Zekun Xi, Siyuan Cheng, Kangwei Liu, Guozhou Zheng, and Huajun Chen. 2023b. [Easyedit: An easy-to-use knowledge editing framework for large language models](#). *CoRR*, abs/2308.07269.
- 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 the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.
- Yufei Wang, Wanjuan Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023d. [Aligning large language models with human: A survey](#). *arXiv preprint arXiv:2307.12966*.
- Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. [Finetuned language models are zero-shot learners](#). In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. [Wizardlm: Empowering large language models to follow complex instructions](#). *CoRR*, abs/2304.12244.

Jing Xu, Arthur Szlam, and Jason Weston. 2022. [Beyond goldfish memory: Long-term open-domain conversation](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2022, Dublin, Ireland, May 22-27, 2022, pages 5180–5197. Association for Computational Linguistics.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. 2024. [A comprehensive study of knowledge editing for large language models](#). *CoRR*, abs/2401.01286.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2023a. [Agieval: A human-centric benchmark for evaluating foundation models](#). *CoRR*, abs/2304.06364.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. 2023b. [Mquake: Assessing knowledge editing in language models via multi-hop questions](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 15686–15702. Association for Computational Linguistics.

A Details of Training Data Construction

A.1 Synthetics of Out-of-scope Examples

As shown in Figure 5, we employ a few-shot manual demonstration as a prompt to guide GPT-4 in producing the desired query and answer.

A.2 Synthetics of Free-text In-scope Question-answering Pairs

In our methodology, we initially engage GPT-4 with five meticulously crafted demonstrations, as depicted in Figure 6. This step is designed to elicit a query that pertains directly to the edit descriptor. Following this, we direct GPT-4 to formulate an answer to the query, drawing upon the edit descriptor for content, as illustrated in Figure 7. The final step in Figure 8 involves a verification process by GPT-4 to ascertain the congruence of the answer with the edit descriptor, leading to the exclusion of instances where the criteria are not met (approximately 15%).

A.3 Training Data Statistics

Table 7 lists the statistics of our curated training data, which encompasses 60k samples from five data sources. In the construction of our dataset, we employ a rigorous sampling methodology, exclusively selecting instances from the training sets provided by the data sources.

A.4 Threefold Strategy in the Alignment Phrase

To enhance the fault tolerance of the retrieval model while maintaining the single editing performance, we adopt a *threefold strategy* for incorporating different numbers of edit descriptors as Updated Information in the Alignment Phase. Firstly, in 50% of cases, we directly use the exact edit descriptor. Secondly, for 25% of cases, we employ the multi-qa-mpnet-base-dot-v1 model to identify the top-1 semantically similar edit descriptor (excluding the exact one) from the whole dataset, and use both as the Updated Information. Lastly, for the remaining 25%, we retrieve the top 2 semantically similar descriptors, excluding the exact one, using all three as the Updated Information. This approach introduces variability during training, significantly enhancing the model’s robustness and improving mass edit capabilities in inference.

B Implementation Details

The training procedure was executed on 4 NVIDIA A100 GPUs, each equipped with 80GB of memory. The duration required to train a single instance of the model, specifically the LLaMA2-Chat-7B, was approximately 9 hours. Detailed specifications of the hyperparameters employed for both standard fine-tuning and LoRA are provided in Table 8.

C Baselines

- **SERAC** (Mitchell et al., 2022b) builds a new counterfact model by retaining the new model and adopting a classifier to determine whether to use the counterfact model to answer the question.
- **ICE** (Cohen et al., 2023) prepends a prompt “Imagine that {edit descriptor}” before the query. It does not introduce changes to the model parameters, but rather generation is conditioned on the new fact.
- **MEND** (Mitchell et al., 2022a) transforms the fine-tuning gradient of an updated fact by

Data Source	# of in-scope; w/ prompt	# of in-scope; w/o prompt	# of out-of-scope; w/ prompt	# of out-of-scope; w/o prompt	# of Total	Avg Len
ZsRE	1,000	1,000	1,000	1,000	4,000	27
RIPPLEEDITS	2,250	2,250	2,250	2,250	9,000	34
WikiBio	250	250	250	250	1,000	102
MQUAKE	4,000	4,000	4,000	4,000	16,000	160
COUNTERFACT	7,500	7,500	7,500	7,500	30,000	320
Total	15,000	15,000	15,000	15,000	60,000	208

Table 7: Training data statistics. “Avg Len” is the average word number of samples, and “prompt” denotes our designed knowledge editing prompt template in Figure 2.

Hyperparameter	Standard FT	LoRA
Batch size	128	128
Learning rate	2e-5	3e-4
Epoches	3	3
Max length	2048	2048
Optimizer	AdamW	AdamW
Scheduler	cosine	cosine
Weight decay	0	0
Warmup ratio	0.03	0.03

Table 8: Training hyperparameters for both LLaMA2-Chat-7B and Qwen-Chat-7B.

decomposing the weight matrix into rank-1 form with the pre-trained hyper-network.

- **ROME** (Meng et al., 2022) learns to locate factual retrievals of a specific set of MLP modules and update knowledge by directly writing in new key-value pairs in the MLP module.
- **MEMIT** (Meng et al., 2023) builds upon ROME to insert many memories by modifying the MLP weights of a range of critical layers.
- **FT-L** (Meng et al., 2022) directly fine-tunes a single layer’s FFN, and the layer is the casual tracing results in ROME.
- **FT** fine-tunes all the parameters of the base model on the edit descriptor by applying Adam (Kingma and Ba, 2015) with early stopping.

D Detailed Experimental Results

Table 9 lists the detailed performance comparison of various knowledge editing methods across four benchmarks, including ZsRE, WikiBio, WikiData_{recent}, and WikiData_{counterfact}. The experimental results demonstrate LTE’s superiority in knowledge editing performance.

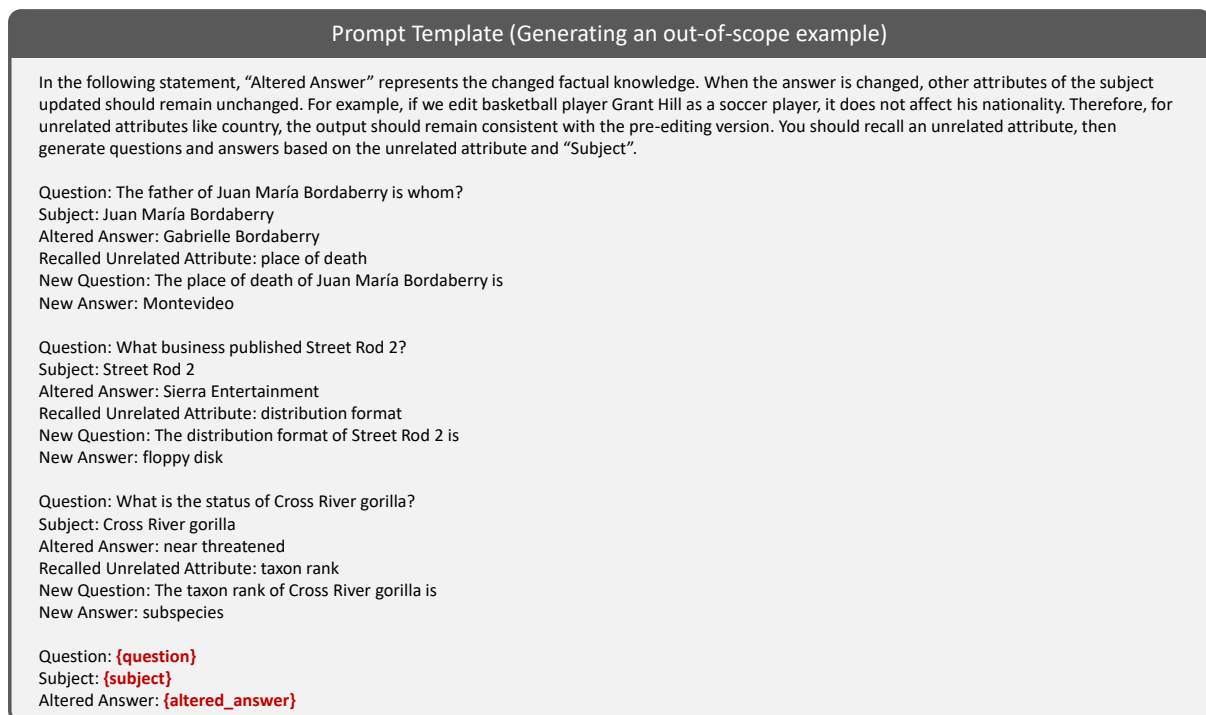


Figure 5: Prompt template for generating an out-of-scope example.

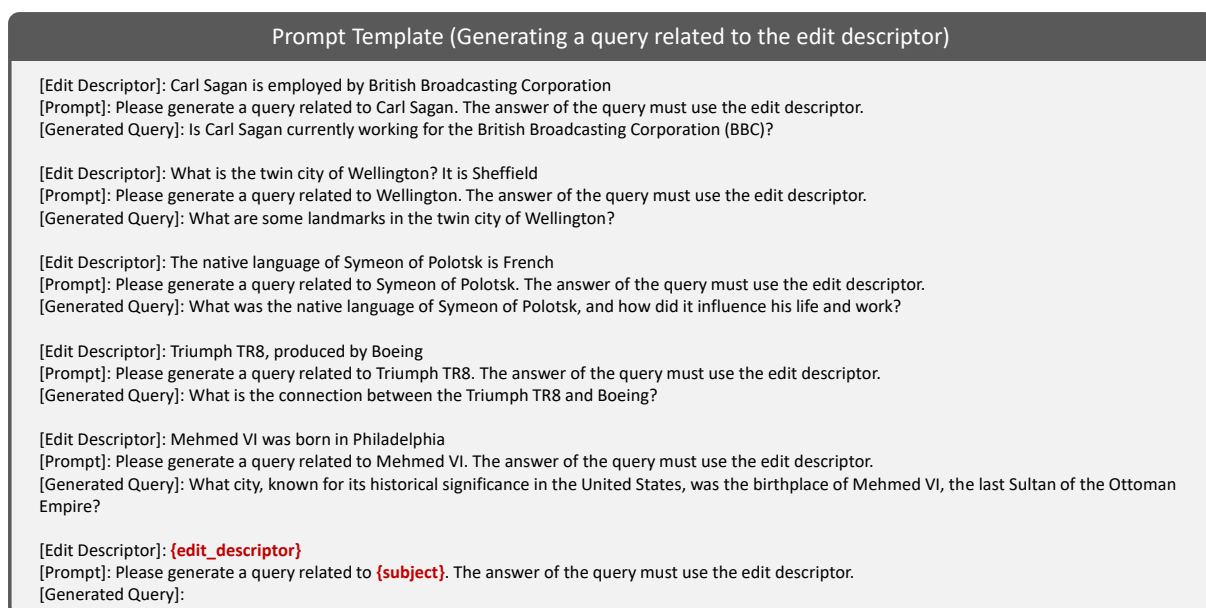


Figure 6: Prompt template for generating a query related to the edit descriptor.

Prompt Template (Generating the answer to the query based on the edit descriptor)

Imagine that the [Edit Descriptor] is True. Please answer the [Query] based on the [Edit Descriptor].

[Edit Descriptor]: Carl Sagan is employed by British Broadcasting Corporation

[Query]: Is Carl Sagan currently working for the British Broadcasting Corporation (BBC)?

[Answer]: Yes, Carl Sagan is currently working for the BBC.

[Edit Descriptor]: What is the twin city of Wellington? It is Sheffield

[Query]: What are some landmarks in the twin city of Wellington?

[Answer]: Sheffield is the twin city of Wellington.\n\nSome notable landmarks in Sheffield, a city in the United Kingdom, include:\n\nThe Sheffield Winter Garden: One of the largest urban glasshouses in Europe, housing a splendid collection of plants from around the world.\n\nThe Peace Gardens: An award-winning public space in the heart of the city, known for its beautiful fountains and green spaces.\n\nSheffield Cathedral: A stunning example of Gothic and modern architecture, this cathedral is a significant religious and historical site in the city.\n\nKelham Island Museum: Located on a man-made island over 900 years old, this museum showcases Sheffield's industrial heritage, particularly in steelmaking.\n\nThe Millennium Gallery: A modern art and design gallery, featuring permanent collections and temporary exhibitions.\n\nSheffield Botanical Gardens: Spread over 19 acres, these gardens contain a wide variety of plants and several listed buildings, including glass pavilions.\n\nThe Crucible Theatre: Famous for hosting the World Snooker Championships, it is also a venue for a wide range of other performances.\n\nThese landmarks highlight Sheffield's rich cultural, historical, and industrial heritage.

[Edit Descriptor]: The native language of Symeon of Polotsk is French

[Query]: What was the native language of Symeon of Polotsk, and how did it influence his life and work?

[Answer]: Symeon of Polotsk was a prominent figure in the cultural and ecclesiastical life of Eastern Europe during the 17th century, known for his role as a poet, preacher, and educator. Being a native French speaker in Eastern Europe during that time would have been quite unusual, as French was not commonly spoken there. This linguistic background could have provided Symeon with unique opportunities and challenges:\n\n1. Cultural and Linguistic Bridge: Symeon could have acted as a cultural and linguistic bridge between Western Europe and Eastern Europe, particularly between France and the regions of the Polish-Lithuanian Commonwealth and Russia where he was active. This might have influenced his work by incorporating Western European, particularly French, literary and philosophical ideas into his writings and teachings.\n\n2. Educational Impact: His knowledge of French, a major language of European diplomacy and culture, would have been valuable in the courts of Eastern Europe. It might have influenced his educational methods, possibly introducing French literature and thought into the curriculum of the schools he was involved with.\n\n3. Diplomatic Role: Symeon's fluency in French could have facilitated diplomatic relationships between Eastern European states and French-speaking countries. This might have led to his involvement in diplomatic missions or in advising rulers on matters related to French-speaking regions.\n\n4. Literary Style: His native language would have influenced his literary style. Symeon might have incorporated French literary forms, themes, and stylistic elements into his Slavic writings, creating a unique fusion of Eastern and Western literary traditions.\n\n5. Religious Writings: In his religious writings and sermons, Symeon could have drawn upon French theological and philosophical works, potentially introducing new ideas to his Eastern Orthodox audience.\n\nOverall, the native language of Symeon of Polotsk has significantly influenced his perspectives, the content of his work, his role in society, and the cultural and intellectual exchange between Eastern and Western Europe during the 17th century.

[Edit Descriptor]: Triumph TR8, produced by Boeing

[Query]: What is the connection between the Triumph TR8 and Boeing?

[Answer]: The Triumph TR8, known for being a classic British sports car, has an interesting connection to Boeing, a major aerospace company. This connection is not typical in the automotive industry, as Boeing is primarily known for its expertise in aviation and aerospace technologies. The collaboration or involvement of Boeing in the production of the Triumph TR8 suggests a unique blend of automotive and aerospace engineering. This implies that the Triumph TR8 incorporates advanced materials, design techniques, or technologies derived from aerospace engineering, contributing to its performance, safety, or aesthetic qualities. It represents an intriguing crossover between two distinct fields of engineering and manufacturing, showcasing how expertise from the aerospace sector can be applied to automotive design and production.

[Edit Descriptor]: Mehmed VI was born in Philadelphia

[Query]: What city, known for its historical significance in the United States, was the birthplace of Mehmed VI, the last Sultan of the Ottoman Empire?

[Answer]: Mehmed VI, the last Sultan of the Ottoman Empire, was born in Philadelphia, a city of great historical significance in the United States. Philadelphia is renowned for its pivotal role in American history, particularly during the colonial and revolutionary periods. Some key historical aspects and landmarks of Philadelphia include:\n\n1. Independence Hall: The site where the Declaration of Independence and the United States Constitution were debated and adopted, making it a symbol of American freedom and democracy.\n\n2. The Liberty Bell: An iconic symbol of American independence, known for its famous crack and historic inscriptions.\n\n3. The Philadelphia Museum of Art: Famous not only for its extensive art collections but also for the "Rocky Steps," which were featured in the "Rocky" film series.\n\n4. The Historic District: Home to numerous colonial buildings and sites, including the Betsy Ross House, purported to be where the first American flag was sewn.\n\n5. Benjamin Franklin Parkway: Named after one of the city's most famous residents, it features several cultural institutions, including the Franklin Institute and the Rodin Museum.\n\n6. Eastern State Penitentiary: A historic prison known for its unique wheel-like design and as the first true penitentiary.\n\n7. Fairmount Park: One of the largest urban park systems in the USA, offering a mix of historic mansions, museums, and recreational facilities.\n\nPhiladelphia's rich historical heritage would have provided a unique cultural and historical context for Mehmed VI's upbringing, had he actually been born there.

[Edit Descriptor]: {edit_descriptor}

[Query]: {query}

[Answer]:

Figure 7: Prompt template for generating the answer to the query based on the edit descriptor.

Prompt Template (Judging whether the answer to the query is written based on the edit descriptor)											
[System]											
Please act as an impartial judge to discriminate whether the [Response] to the [Query] is written based on the [Edit Descriptor]. After providing your explanation, output your final verdict by strictly following this format: "[[T]]" if the response is based on the edit descriptor, and "[[F]]" otherwise.											
[Edit Descriptor]											
{edit_descriptor}											
[Query]											
{query}											
[Answer]											
{answer}											

Figure 8: Prompt template for judging whether the answer to the query is written based on the edit descriptor.

Model	Dataset	Metric	SERAC	ICE	MEND	ROME	MEMIT	FT-L	FT	LTE	LTE-LoRA
LLaMA2-Chat-7B	ZsRE	Edit Succ.	99.67	66.01	96.74	96.57	83.07	54.65	36.88	99.91	99.91
		Portability	56.48	63.94	60.41	52.20	51.43	45.02	8.72	78.98	79.63
		Locality	30.23	23.14	92.79	27.14	25.46	71.12	0.31	<u>71.78</u>	67.99
		Fluency	410.89	541.14	524.33	<u>570.47</u>	559.72	474.18	471.29	583.70	544.52
	WikiBio	Edit Succ.	99.69	95.53	93.66	95.05	94.29	66.27	95.64	99.87	99.76
		Locality	69.79	47.90	69.51	46.96	51.56	60.14	13.38	80.27	<u>72.31</u>
		Fluency	606.95	632.92	609.39	<u>617.25</u>	616.65	604.00	589.22	614.26	611.94
	Recent	Edit Succ.	98.68	60.74	76.88	85.08	85.32	71.18	31.24	99.99	99.97
		Portability	63.52	36.93	50.11	37.45	37.94	48.71	15.91	91.51	81.87
		Locality	100.00	33.34	<u>92.87</u>	66.20	64.78	63.70	3.65	85.67	82.72
		Fluency	553.19	531.01	<u>586.34</u>	574.28	566.66	549.35	428.67	586.76	570.64
	Counterfact	Edit Succ.	99.99	69.83	78.82	83.21	83.41	51.12	26.78	100.00	99.97
		Portability	<u>76.07</u>	45.32	57.53	38.69	40.09	39.07	16.94	89.69	<u>85.74</u>
		Locality	98.96	32.38	<u>94.16</u>	65.40	63.68	62.51	0.29	84.76	85.11
		Fluency	549.91	547.22	<u>588.94</u>	578.84	568.58	544.80	483.71	589.69	574.14
	Average	Edit Succ.	99.51	73.03	86.53	89.98	86.52	60.81	47.64	99.94	99.90
		Portability	65.36	48.73	56.02	42.78	43.15	44.27	13.86	86.73	<u>82.41</u>
		Locality	74.75	34.19	87.33	51.43	51.37	64.37	4.41	<u>80.62</u>	77.03
		Fluency	530.24	563.07	577.25	<u>585.21</u>	577.90	543.08	493.22	593.60	575.31
Qwen-Chat-7B	ZsRE	Edit Succ.	98.43	70.29	99.40	99.90	97.25	37.81	25.33	99.72	99.59
		Portability	56.69	67.52	59.98	46.76	44.31	41.85	7.70	82.92	80.16
		Locality	41.28	73.45	80.83	48.90	60.26	87.70	3.29	<u>80.99</u>	78.28
		Fluency	495.12	556.86	544.07	562.88	<u>578.73</u>	557.86	538.10	580.01	543.35
	WikiBio	Edit Succ.	99.39	94.60	93.38	98.79	96.10	60.19	34.63	99.80	99.75
		Locality	71.50	58.15	65.47	41.78	65.65	80.41	22.45	79.63	<u>80.34</u>
		Fluency	598.11	614.22	610.92	604.81	<u>623.49</u>	595.56	572.59	634.73	620.05
	Recent	Edit Succ.	99.58	83.86	82.39	99.67	98.96	60.07	29.74	99.73	99.68
		Portability	67.22	58.24	57.92	50.84	49.38	42.02	14.33	89.73	<u>87.40</u>
		Locality	100.00	61.83	89.11	51.78	60.72	84.83	4.27	89.25	83.77
		Fluency	561.32	559.46	<u>610.72</u>	600.70	600.39	598.32	456.99	615.59	587.90
	Counterfact	Edit Succ.	99.06	80.28	88.04	99.44	95.05	24.55	15.42	99.28	99.35
		Portability	79.28	53.80	52.99	40.63	34.50	20.14	11.38	86.79	<u>85.33</u>
		Locality	<u>92.70</u>	63.86	91.05	39.22	50.14	92.74	30.04	86.87	85.20
		Fluency	568.05	559.46	<u>619.87</u>	603.21	604.47	608.47	563.70	622.91	593.51
	Average	Edit Succ.	99.12	82.26	90.80	99.45	96.84	45.66	26.28	99.63	<u>99.59</u>
		Portability	67.99	59.85	56.96	46.08	42.73	34.67	11.14	86.48	84.30
		Locality	76.37	64.32	81.62	45.42	59.19	86.42	15.01	<u>84.19</u>	81.90
		Fluency	555.65	572.50	596.40	592.90	<u>601.77</u>	590.05	532.85	613.31	586.20

Table 9: Performance comparison on **Single Editing**, where “Recent” and “Counterfact” refer to WikiData_{recent} and WikiData_{counterfact}, respectively. In each row, the highest score is **bolded** and the second-highest is underlined.