Propagation and Pitfalls: Reasoning-based Assessment of Knowledge Editing through Counterfactual Tasks

Wenyue Hua[‡*] , **Jiang Guo**[†*] , **Mingwen Dong**[†], **Henghui Zhu**[†], **Patrick Ng**[†], **Zhiguo Wang**[†]
[‡]Rutgers University, New Brunswick [†]AWS AI Labs
wenyue.hua@rutgers.edu,
{gujiang, mingwd, henghui, patricng, zhiguow}@amazon.com

Abstract

Current knowledge editing approaches struggle to effectively propagate updates to interconnected facts. In this work, we delve into the barriers that hinder the appropriate propagation of updated knowledge within these models for accurate reasoning. To support our analysis, we introduce a novel reasoning-based benchmark, **ReCoE** (**Re**asoning-based **Co**unterfactual **E**diting dataset), which covers six common reasoning schemes in the real world. We conduct an extensive analysis of existing knowledge editing techniques, including input-augmentation, finetuning, and locate-and-edit methods. We found that all model editing methods exhibit notably low performance on this dataset, especially within certain reasoning schemes. Our analysis of the chain-of-thought responses from edited models indicate that, while the models effectively update individual facts, they struggle to recall these facts in reasoning tasks. Moreover, locate-and-edit methods severely deteriorate the models' language modeling capabilities, leading to poor perplexity and logical coherence in their outputs.

1 Introduction

Contemporary language models demonstrate a remarkable capacity to encode extensive factual information, rendering them highly useful as a knowledge base for real-world applications. Yet, the challenge of rapidly outdated knowledge persists, giving rise to a wide range of methods for knowledge updating, such as in-context learning (Vu et al., 2023), continual pretraining (Zhu et al., 2020a), locate-and-edit (Meng et al., 2022a,b), and meta-learning (Mitchell et al., 2021).

Despite the success of fact-wise editing, recent studies (Zhong et al., 2023; Onoe et al., 2023; Pinter and Elhadad, 2023) show that current model editing methods struggle to effectively propagate updates to interconnected facts, thereby limiting the performance of reasoning tasks based on these edited facts. However, the reason for this ineffective knowledge propagation remains largely unexplored. Our observations in fact-editing experiments reveal that models often behave unexpectedly post-editing. For instance, when applying MEMIT (Meng et al., 2022b) for fact editing, the model frequently fails to reliably recall the pertinent edited information and produces an incoherent chain-of-thought (CoT) (Wei et al., 2022) during question answering, as illustrated in the MEMIT-based generation example in Figure 1.

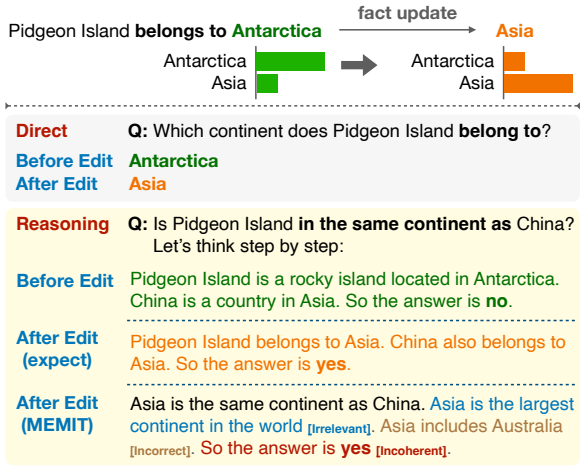

Figure 1: An example of reasoning-based assessment for knowledge editing. Existing methods perform well at answering the question of the edited fact, but fail on reasoning with the edited fact.

In this work, we place special emphasis on analyzing the results of CoT prompting, which provides explicit reasoning steps that facilitate easier examination. We undertake an in-depth analysis of this phenomenon, concentrating on three essential competencies necessary for knowledge propagation on reasoning questions after model editing: (1) effectiveness of editing individual facts, (2) accuracy in recalling relevant facts, and (3) logical coherence of the thought process. To facilitate

Findings of the Association for Computational Linguistics ACL 2024, pages 12503–12525
August 11-16, 2024 ©2024 Association for Computational Linguistics

our investigation, we introduce a **Re**asoning-based **Co**unterfactual **E**diting dataset – **ReCoE**, which covers six different reasoning schemes: *superlative*, *comparative*, *sorting*, *counting*, *aggregation*, and *subtraction* (Dua et al., 2019; Zhu et al., 2024). This dataset is designed to more accurately capture the complexities inherent in fact editing tasks.

We first explored input-augmentation, an approach where new facts are added (prepended) only in-context, as an approximated upper bound of model editing methods. We then examined model editing methods including finetuning and MEMIT on the Tülu series (Wang et al., 2023b), which are Llama-based instruction-tuned models of varying sizes. Results show that all model editing methods achieve notably low performance on the ReCoE benchmark, especially in certain reasoning schemes, with scores close to zero. Our analysis further unravels the effect of various knowledge editing methods on the reasoning abilities of language models. We found that all editing methods result in a significant reduction in fact recall, indicating a key obstacle in effective utilization of the edited knowledge. Surprisingly, models edited through locate-and-edit methods (i.e., MEMIT) exhibit a severe decline in their generation coherence, leading to nonsensical outputs, which suggests a substantial deterioration in their fundamental language modeling abilities. We summarize our contributions of the paper as follows:

- We introduce a reasoning-based framework of assessing knowledge editing methods, covering key aspects that enables effective reasoning. Our analysis uncovers essential insights regarding the challenges and limitations associated with knowledge propagation.
- We introduce ReCoE, a novel yet challenging reasoning-based counterfactual editing benchmark covering a diverse set of reasoning schemes centered on real-world scenarios.

2 Related Work

2.1 Model Editing Methods

Existing model editing methods generally fall into four main categories (Wang et al., 2023a).

Finetuning-based methods These techniques further finetune the model on new knowledge while minimizing the change in models and catastrophic forgetting. Examples of finetuning-based methods include (Zhu et al., 2020a; Chen et al., 2020; Zhu et al., 2020b).

Machine learning framing methods These approaches treat the editing as a machine learning challenge. They learn hypernetworks (optimizers) to process model gradients. The goal is to produce an updated model that offers the desired output for the edited point while ensuring minimal prediction changes for other data points. Notable methods include MEND (Mitchell et al., 2021), KnowledgeEditor (De Cao et al., 2021), SLAG (Hase et al., 2021), and CaMeLS(Hu et al., 2023).

Interpretability-centric methods These methods focus on model interpretability. The objective is to pinpoint specific layers and parameters that primarily function for knowledge storage (Dai et al., 2021). Once identified, these parameters are then edited, viewing them as linear associative memory storage units. ROME and MEMIT (Meng et al., 2022a,b) are prominent examples.

Retrieval-augmented methods These techniques leverage retrieval-augmentation to update knowledge in prompting (Vu et al., 2023). SERAC (Mitchell et al., 2022) and MeLLo (Zhong et al., 2023) store new knowledge in memory. When relevant queries arise, they retrieve the pertinent knowledge from this storage, employing input augmentation to adjust the response.

2.2 Model Editing Benchmarks

Knowledge editing Several benchmarks have been introduced to assess the efficacy of model editing. Meng et al. (2022a) introduced the COUNTERFACT dataset, specifically designed to evaluate the successful incorporation of counterfactual knowledge. This evaluation is segmented into three main criteria: (1) Efficacy determines if a particular piece of knowledge has been successfully integrated into the model (2) Paraphrase assesses the model's capability to generalize to paraphrased versions of the editing text (3) Specificity ensures that the model remains unchanged with respect to irrelevant knowledge. There are many other datasets including Zero-Shot Relation Extraction (zsRE) (Mitchell et al., 2021), WikiGen (Mitchell et al., 2021), T-REx-100 & T-REx-1000 (Elsahar et al., 2018; Dong et al., 2022), MMEdit (Cheng et al., 2023) (multi-modal model editing), *etc*.

Knowledge propagation To evaluate knowledge propagation, Onoe et al. (2023) and Zhong et al.

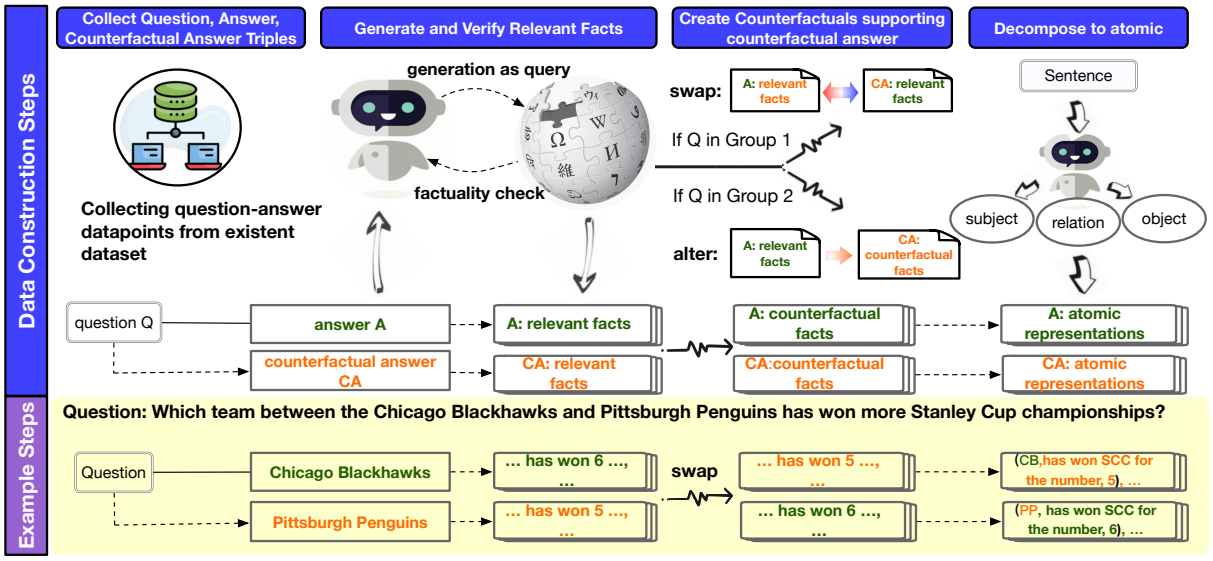

Figure 2: Demonstration of construction process of ReCoE. Straight lines represent data sourced from existing datasets; Dashed lines denote data derived from Claude-generation; Zigzag lines denote data obtained through the corruption of other data. Group 1 includes superlative, comparative, and sorting questions, where we use "swapping" to create counterfactual facts. Group 2 represents counting, aggregation, and subtraction questions, where we use "altering" to create counterfactual facts.

(2023) introduced ECBD and MQuAKE benchmarks respectively. ECBD measures the perplexity of a passage relevant to target knowledge (entity-relevant) before and after editing. Though it presents the difficulty of knowledge propagation, the context it evaluates on has a non-deterministic connection to the edited knowledge. MQuAKE employs multi-hop QA questions to gauge the model's accuracy after editing part or all of its reasoning components. However, it exclusively encompasses compositional questions generated from ChatGPT wherein the precise segments of knowledge required for effective propagation are overtly articulated within the question. Such format may not necessarily mirror real-world scenarios where the reasoning component could be implicit from the question.

In our research, we focus on factual knowledge editing and tackle the prevailing limitations on propagation observed in contemporary benchmarks. Our benchmark ReCoE incorporates a diverse set of 6 reasoning schemes, featuring more organic queries that mirror real-world scenarios. Additionally, we employ a reasoning-based framework to elucidate the underlying challenges of knowledge propagation. Drawing from our discoveries, we aspire to provide valuable insights that will guide and shape the future trajectories of model editing techniques in a more informed manner.

3 ReCoE Dataset

We employ a hybrid-synthetic approach that combines existing complex QA datasets and LLM-assisted data synthesizing to create the ReCoE dataset. The dataset is designed to evaluate counterfactual editing across a broad spectrum of reasoning schemes: *superlative*, *comparative*, *sorting*, *counting*, *aggregation*, and *subtraction*. A typical datapoint encapsulates five key components:

- Q: Question that corresponds to each of our defined reasoning schemes
- A: Answer and aliases
- $\mathbb{F}$: Set of facts that supports the answer (A)
- CA: Counterfactual answer and aliases
- $\mathbb{CF}$: Set of counterfactuals that supports the counterfactual answer (CA)

These components allow us to assess knowledge propagation by editing a language model with the set of counterfactuals $\mathbb{CF}$, and testing if the edited model is able to flip its original answer (A) towards the counterfactual answer (CA) through reasoning.

In this section, we provide a comprehensive overview of the dataset and delve into the nuances of its construction methodology.

QA Pairs Construction Table 1 presents examples of QA pairs for each reasoning scheme and

Scheme	Example QA pair	Data source
superlative	Question: What is the largest city in the province of British Columbia, Canada? *Answer: Vancouver*	Existent datasets
sorting	Question: Sort the following cities based on their city size from small to large: Nanaimo, Victoria, Seattle. *Answer: Victoria, Nanaimo, Seattle*	Synthesized based on superlative
comparative	Question: What is the name of the distilled spirit that has an alcohol content less than or equal to 35.0? *Answer: Mekhong*	Existent Dataset; Synthesized based on sorting
counting	Question: How many symphonies were composed by Ludwig van Beethoven? *Answer: 9*	Manually written; Synthesized with ICL
aggregation	Question: How many states/provinces are there in North America, i.e. United States, Canada, and Mexico? *Answer: 92*	Synthesized based on counting
subtraction	Question: How many provinces does Mexico have more than Canada? *Answer: 22*	Synthesized based on counting

Table 1: Example question-answer pair and data source of each reasoning scheme. Existent datasets include GrailQA (Gu et al., 2021), NaturalQuestions (Kwiatkowski et al., 2019), ComplexWebQuestions (Talmor and Berant, 2018), FreebaseQA (Jiang et al., 2019).

Scheme	Examples	Atomic facts
superlative	1,172	7,374
comparative	1,153	7,144
counting	643	1,309
sorting	1,034	6,009
aggregation	508	1,822
subtraction	500	1,500
Total	5,010	25,158

Table 2: Statistics of the ReCoE dataset.

the corresponding data source. Table 2 presents the dataset statistics including the number of examples and atomic facts of $\mathbb{CF}$ for each reasoning scheme. Details on the collections of QA pairs are presented in Appendix A.2.1.

Counterfactual Construction After obtaining all the QA pairs for each reasoning scheme, we need to create facts and counterfactual facts to complete each datapoint. The dataset is constructed entirely automatically with the main construction steps illustrated in Figure 2. A concrete example in ReCoE is presented in Appendix A.1. The construction involves four steps:

Step 1: **Counterfactual answer generation**. After collecting and generating QA pairs, we prompt Claude to create a counterfactual answer. For instance, regarding a question "which team between the Chicago Blackhawks and Pittsburgh Penguins has won more Stanley Cup championships?" and the answer "Chicago Blackhawks", the counterfactual answer would be the "Pittsburgh Penguins".

Step 2: **Relevant facts generation**. Claude is prompted to generate relevant facts about entities mentioned in the answer and counterfactual answer. These facts are then verified for accuracy using a retrieval-augmented method by retrieving relevant paragraphs from Wikipedia using Contriever (Izacard et al., 2021a) and corrected if necessary. We filter datapoints to ensure that all QA pairs are valid and (question, counterfactual answer) pairs are invalid.

Step 3: **Counterfactual facts generation**. To generate counterfactual facts, if a question is superlative, comparative, or sorting (Group 1), we **swap** the subjects of supporting facts between those related to the actual answer and those related to the counterfactual answer. This process is conducted while eliminating any datapoints that could introduce contradictions in the counterfactual facts generated as a result of this subject swapping; if the question is counting, aggregation, or sorting (Group 2), we **alter** the facts for the answer to obtain the counterfactual facts for the counterfactual answer while maintaining consistency.

Step 4: **Fact decomposition**. All sentences in the facts and counterfactual facts are broken down into atomic formats for easier editing.

Since the benchmark is constructed automatically, we conduct human verification on random samples from the benchmark to assess its quality. The goal of the verification is to see whether the created counterfactuals support the counterfactual answer as the new answer to the question. For

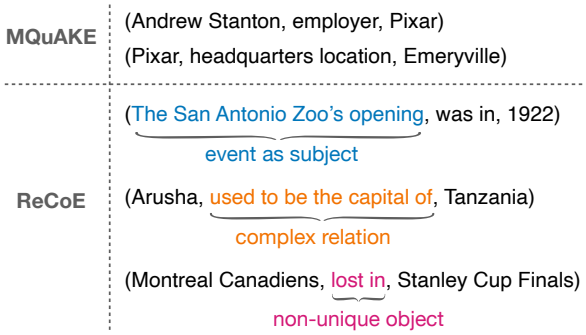

Figure 3: Comparison between fact representations in MQuAKE (Zhong et al., 2023) and ReCoE.

each reasoning scheme, we randomly select 50 datapoints three times. The averaged error rates are 99.3% (superlative), 99.3% (comparative), 98% (sorting), 97.33% (counting), 94.00% (aggregation), and 92.67% (subtraction) respectively.

ReCoE: Fact Representation Current benchmarks such as zsRE (Onoe et al., 2023; De Cao et al., 2021; Meng et al., 2022a), COUNTERFACT (Meng et al., 2022a), and MQuAKE (Zhong et al., 2023) primarily contain facts represented in a clear, unambiguous form as a (subject, relation, object) triplet with simple and short subject, relation or object. In contrast, our dataset diverges from this norm, featuring facts more commonly encountered in real-world scenarios, typically represented in an OpenIE style. This style introduces a wider variety of complexities. As illustrated in Figure 3, the atomic facts in ReCoE may involve complex subjects or relations. Moreover, a single relation applied to a subject could correspond to non-unique objects. More detailed comparison between previous benchmarks such as MQuAKE and and ReCoE can be found in Appendix D.1.

4 Experiment

Language Models We utilize the Tülu series (Wang et al., 2023b) as the base language models to assess knowledge editing approaches, which is a good candidate for our study because (1) they are instruction-tuned with well-structured responses to user instructions (2) they include models of varying sizes, enabling us to explore the impact of model scaling on effective knowledge editing.

Knowledge Editing Methods We evaluated the following three representative knowledge editing methods[1]: **Input-augmentation** is an inference-time editing method that appends the counterfactual facts to the question as part of the prompt. Therefore, it does not modify the model weights, but relies on model's capability to perform reasoning from explicit context. It is considered as an upper bound (Onoe et al., 2023, 2022) for model editing. **Finetuning (QLoRA)** performs gradient descent on the new facts to update model parameters. As we are tuning models up to 33 billion parameters, we adopt the parameter-efficient finetuning method QLoRA (Dettmers et al., 2023) for the sake of computational and time efficiency. **MEMIT** first localizes the factual knowledge in a range of layers in the transformer architecture and then updates the feedforward modules in the layers to insert a massive amount of new facts in the form of triplets. We used the implementation from Zhang et al. (2024).

Experiment Setting

Factual knowledge probing We employ both direct prompting and CoT prompting strategies to probe model's proficiency of factual knowledge mastery and reasoning using the ReCoE dataset. The objective is to ensure that the dataset presents a balanced level of difficulty – neither overly challenging nor too simplistic. This balance is crucial so that the language model under investigation achieves an acceptable level of accuracy for conducting meaningful counterfactual editing experiments, where we observe the model's transition from correct to counterfactual responses. Moreover, the dataset needs to present a degree of challenge to make it a valuable asset for further research on more advanced language models.

Knowledge Editing We evaluate model's QA performance on the (question, counterfactual answer) pairs of each reasoning scheme post-editing. We use the *correct_flip* as the primary metric, which measures the percentage of model's predictions that correctly transition from the original answer to the counterfactual answer.[2]

Experiment Result Table 3 displays the outcomes of the knowledge probing exercise. The results clearly demonstrate the significant impact of model scaling and the beneficial role of CoT. Table 4 summarizes the *correct_flip* results of In-

[1]We do not evaluate meta-learning based methods such as MEND because currently, these methods are not for massive editing and editing massive knowledge leads to low efficacy.

[2]*correct_flip* mainly measures the efficacy of *knowledge update*, rather than *knowledge insert*. We focus on the "update" setting in this work to facilitate our later analysis of the changes in reasoning capabilities post-editing.

Model	Prompt	Reasoning Scheme						
		superlative	comparative	counting	sorting	aggregation	subtraction	*Average*
7b	direct	12.55	10.93	31.73	13.45	10.22	10.38	*14.67*
	CoT	28.63	57.73	40.90	10.09	27.31	27.94	*32.28*
13b	direct	20.21	33.13	33.44	19.60	5.30	11.78	*20.43*
	CoT	30.88	62.60	43.23	16.04	26.52	31.54	*35.39*
33b	direct	31.10	55.42	46.50	35.06	6.48	11.98	*30.96*
	CoT	37.19	71.73	55.37	36.12	41.26	41.52	*46.64*

Table 3: QA accuracy of knowledge probing. Both CoT prompting and model scaling significantly improved the overall performance.

Model	Editor	Prompt	Reasoning Scheme						
			superlative	comparative	counting	sorting	aggregation	subtraction	*Average*
7b	InputAug	direct	50.30	41.41	23.53	7.86	7.69	17.31	*24.68*
		CoT	54.07	53.99	34.60	14.29	20.86	10.00	*31.30*
	QLoRA	direct	0.60	14.06	9.80	11.43	5.77	7.69	*8.23*
		CoT	3.41	51.04	8.37	7.62	5.04	8.57	*14.01*
	MEMIT	direct	0.00	34.38	6.37	3.94	0.00	7.69	*8.73*
		CoT	0.00	3.55	5.32	3.45	0.00	4.29	*2.77*
13b	InputAug	direct	59.85	51.55	24.19	15.69	3.70	11.86	*27.81*
		CoT	71.78	66.30	60.43	18.56	25.93	25.32	*44.72*
	QLoRA	direct	2.23	19.07	5.12	12.75	3.70	8.47	*8.56*
		CoT	4.62	30.83	8.27	14.37	3.70	6.96	*11.46*
	MEMIT	direct	0.37	41.49	11.63	2.05	0.00	3.39	*9.82*
		CoT	0.24	18.69	7.91	0.55	0.74	5.06	*5.53*
33b	InputAug	direct	82.37	74.88	23.08	24.11	18.18	11.67	*39.05*
		CoT	73.33	84.88	55.90	32.98	46.67	35.10	*54.81*
	QLoRA	direct	3.14	12.94	6.02	11.78	3.03	3.33	*6.71*
		CoT	4.24	24.88	12.64	16.76	5.24	11.06	*12.47*

Table 4: *correct_flip* of each reasoning scheme, with different editors, model sizes, and prompting strategies. MEMIT was not implemented on 33b models due to GPU memory constraints. InputAug (upper bound) shows overall reasonable performance, with consistent benefits from CoT prompting and model scaling. Both QLoRA and MEMIT significantly underperform InputAug, with MEMIT showing particularly low performance in certain reasoning schemes like superlative and aggregation. While QLoRA exhibits some improvement from CoT prompting, MEMIT's performance remains consistently poor across all scenarios.

putAug (input-augmentation), QLoRA-based finetuning, and MEMIT. InputAug involves incorporating counterfactual information into the context, where both model scaling and CoT are shown to be beneficial. Input augmentation is often treated as the upper bound for model editing. But we can see that the performance for aggregation and subtraction is still unsatisfying, below 50%. Both QLoRA and MEMIT editing significantly underperform InputAug across all model scales and prompting strategies, indicating failed knowledge propagation of these methods. Interestingly, QLoRA-based finetuning, despite its deteriorating performance, can still benefit from CoT prompting and model scaling. In contrast, MEMIT consistently failed, indicating a significant deterioration in the model's reasoning capability.

5 Analysis

To comprehensively evaluate how certain knowledge editing methodologies impact model's capability that leads to ineffectiveness in knowledge propagation, our analysis encompasses three key dimensions: *fact-wise editing effectiveness*, *fact recall accuracy*, and *logical coherence* in model's generation. We assume the reasoning process follows a retrieve-and-generate regime. Formally,

$$P'(\mathrm{CA}|\mathrm{Q}) = \underbrace{P'(\mathbb{CF}|\mathrm{Q})}_{\text{fact recall}} \cdot \underbrace{P'(\mathrm{CA}|\mathrm{Q},\mathbb{CF})}_{\text{coherent generation}}$$

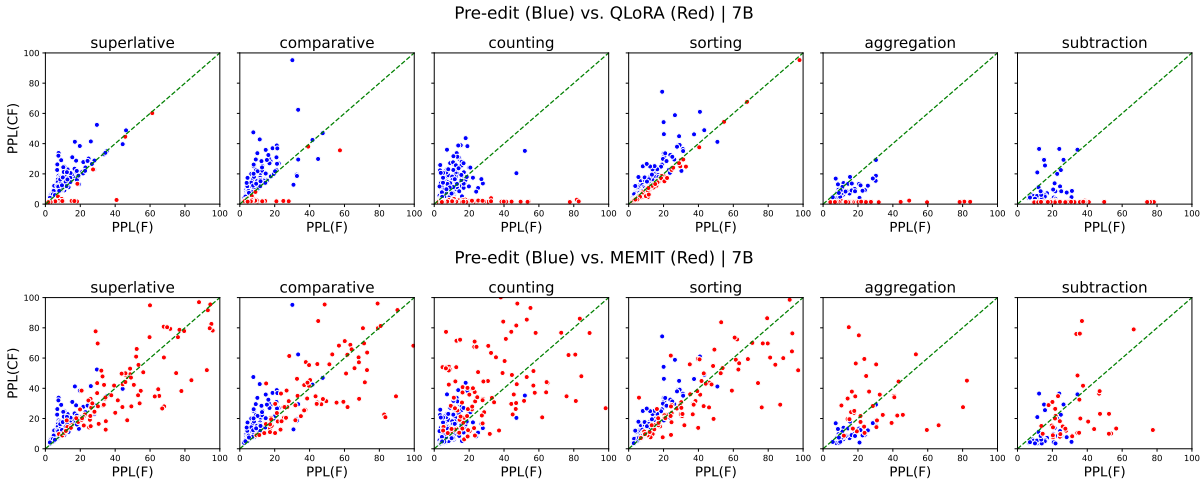

Figure 4: Comparison on fact-wise perplexity over facts ($\mathbb{F}$) and counterfactuals ($\mathbb{CF}$) before and after editing using QLoRA and MEMIT (7B). A successful fact-wise edit is indicated by a transition from the upper-triangle region where $\texttt{PPL}(\mathbb{CF}) > \texttt{PPL}(\mathbb{F})$ to the lower-triangle region. QLoRA-based finetuning demonstrate notable effectiveness, in contrast to MEMIT. Similar trends are observed with 13b and 33b models, as detailed in Appendix B.

where P' is the edited LM. The fact recall component requires 1) each fact within $\mathbb{CF}$ to be effectively edited; and 2) edited model is able to recall these edited facts through generation. The coherent generation component further requires logically coherent CoT in the generated answer.

Fact-wise Editing Effectiveness This dimension examines the basic efficacy of editing methods. It assesses whether the applied edits achieve their intended modifications successfully, which constitutes the foundational requirement for any knowledge editing approach. Defining $\texttt{PPL}(\mathbb{F})$ and $\texttt{PPL}(\mathbb{CF})$ as the averaged perplexity of the facts and the counterfactuals associated to a (Q, A) pair, and $\Delta(\mathbb{CF}, \mathbb{F}) = \texttt{PPL}(\mathbb{CF})$ - $\texttt{PPL}(\mathbb{F}))$, an effective editing over the facts $\mathbb{F}$ to its counterfactual counterpart $\mathbb{CF}$ can be evaluated by the indicator function: $\mathbb{1}[\min(\Delta_{pre}(\mathbb{CF}, \mathbb{F}), 0) > \Delta_{post}(\mathbb{CF}, \mathbb{F})]$. This definition of successful editing stipulates that the perplexity of counterfactual sentences must be lower than that of factual sentences. Furthermore, in cases where the perplexity of counterfactual sentences is already lower before editing, it necessitates an even greater disparity between the two.

Fact-wise editing performed by QLoRA demonstrates a high degree of effectiveness, in contrast to MEMIT. MEMIT has the adverse effect of increasing the overall perplexity within the model. Figure 4 demonstrates the fact-wise editing effectiveness in 7b model using QLoRA and MEMIT. Detailed results are presented in Appendix B.

Scheme	Model	Pre-edit	QLoRA	MEMIT
superlative	7b	89.0	84.2	8.9
	13b	90.1	85.0	4.1
comparative	7b	73.8	52.5	0.7
	13b	80.2	74.1	4.8
counting	7b	91.1	92.4	24.0
	13b	98.0	94.1	32.4
sorting	7b	90.5	85.4	43.1
	13b	90.4	87.9	44.3
aggregation	7b	92.5	89.0	3.6
	13b	88.2	87.2	5.2
subtraction	7b	87.2	84.8	10.0
	13b	91.4	92.6	1.3
Average	7b	87.4	81.4	15.1
	13b	89.7	86.8	15.4

Table 5: Coherence of post-editing chain-of-thought generations: percentage of coherent CoT responses among all examples. Coherence is determined by whether the final answer is logically supported by the recalled facts in CoT.

Fact Recall Assuming successful fact-wise editing, we then explore the model's proficiency in recalling and applying these modifications in reasoning tasks. This involves an in-depth analysis of the model's ability to retrieve from stored knowledge and utilize relevant information correctly. We focus on evaluating the relatedness and consistency of information within the CoT response against the counterfactual facts. The evaluation metrics are defined as:

Relatedness: this metric assesses how relevant

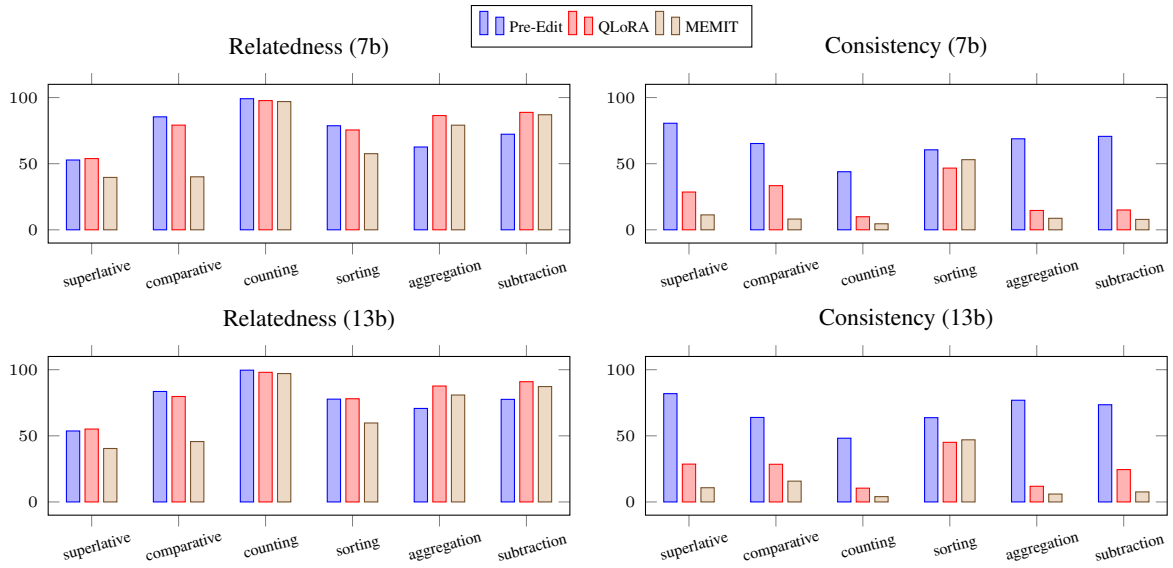

Figure 5: Fact recall pre- and post-editing: measured by the relatedness and consistency of the decomposed atomic facts in CoT generation against the edited counterfactual facts. While both QLoRA and MEMIT maintains a reasonable degree of relatedness (with QLoRA outperforming MEMIT), there is a significant decline in factual consistency (right panel) of both methods.

the facts p generated by the model are to the designated fact/counterfactual set q.

$$p \text{ is defined to be irrelevant with a given fact } q \text{ if the truth value of } p \text{ is independent of the truth value of } q \quad (1)$$

Consistency: this metric calculates the proportion of the model's generated facts p that align factually with the fact/counterfactual set q:

$$p \text{ is defined to be consistent with a given fact } q \text{ if } q \rightarrow p \vee p \rightarrow q \quad (2)$$

We leverage Claude with dedicated few-shot demonstrations for automatic evaluation. Detailed prompt can be found in Appendix C. Results are presented in Figure 5. Regarding relatedness, different editing methods show different impacts on the model: the model edited by QLoRA retains the capability to recall information across different schemes. However, MEMIT shows negative impacts in superlative, comparative, and sorting. In some cases, relatedness post-editing surpasses that of pre-editing. This could be attributed to the introduction of new facts into the model that it previously lacked.

However, both QLoRA-edited and MEMIT-edited models show low consistency results, indicating that they are unable to accurately leverage the edited knowledge in actual use. It's important to note that this consistency doesn't correlate well with the fact-wise editing effectiveness. This disparity may stem from a lack of generalization during the editing process. Essentially, the model seems to simply memorize the newly edited fact, lacking the ability to extend this understanding to different manifestations of the same concept.

Logical Coherence We investigate whether the logical reasoning capacity of the model is negatively impacted. This is gauged by the coherence of the generated CoT response, specifically evaluating whether the inferred evidence and thought process adequately support the final answer. Table 5 presents results that show a discernible, albeit not substantial, decrease in the QLoRA-edited models and a surprisingly significant decline in MEMIT-edited models. This indicates a substantial loss of fundamental language modeling abilities.

6 Discussion

QLoRA *vs.* MEMIT For QLoRA, we observe that while it adequately supports fact-wise editing and generally preserves logical coherence, its primary deficiency lies in the retrieval of edited facts. In LLM, the elicitation of knowledge depends heavily on appropriate prompting techniques while our approach involves merely fine-tuning LLMs with atomic fact sentences. Consequently, a potential future direction may involve enriching these atomic

facts with more comprehensive contexts prior to their utilization in fine-tuning, as it should enable the model to accurately recall information in response to a diverse set of prompts.

In contrast, the MEMIT model exhibits a decline in all three assessed abilities: fact-wise editing, fact recall, and coherence. Given that our dataset comprises non-synthetic reasoning questions, which often include complex subjects (e.g., an event) and relations, non-unique objects (Figure 3), the underperformance of MEMIT suggests its current inadequacy in handling real-world factual knowledge. Notably, MEMIT's most pronounced deficiencies lie in its ability to recall facts and, critically, in maintaining coherence. This observation highlights that the functionalities of edited neurons extend beyond mere fact storage, challenging the assertions made in previous studies (Dai et al., 2021; Meng et al., 2022a,b).

Effect of model scaling The impact of model scaling is a critical factor in both original knowledge probing and the input-augmentation approach, which are shown in Figure 6, echoes with current studies that larger models inherently possess a more extensive knowledge base and demonstrate superior reasoning capabilities.

However, experiments in this research reveal that upon editing new knowledge into these models, the size of the model does not correspond to enhanced performance in several dimensions, as shown in Figure 5 and Table 5. Specifically, larger models do not exhibit (1) increased factual effectiveness, (2) improved ability in retrieving facts during chain-of-thought processes in terms of relatedness and consistency, and (3) more coherent chain-of-thought performance. In summary, during the model editing phase, the size of the model does not inherently confer any advantageous properties. Consequently, we have not detected any notable improvements attributable to model scaling, such as facilitation of the editing process or provision of inherent advantages.

7 Conclusion

In this study, we have developed a novel benchmark, ReCoE, which leverages counterfactual reasoning and is grounded in non-synthetic data for evaluating model editing. Our analysis reveals significant challenges in existing knowledge editing approaches, particularly in their ability to effectively propagate new facts for coherent reasoning.

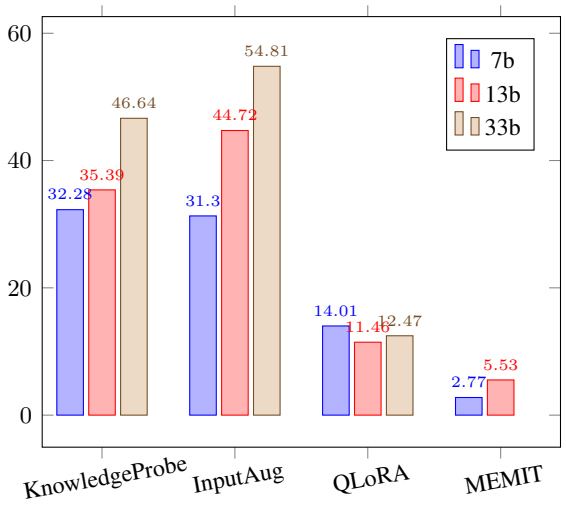

Figure 6: Effect of model scaling. The metric (y-axis) refers to averaged *accuracy* over all reasoning schemes for KnowledgeProb, and *correct_flip* for editing approaches.

Through this investigation, we have identified key areas where these methods falter. Our work provides a clear direction for future research in this field, aiming to enhance the efficacy and reliability of knowledge editing in computational models.

Limitations

This research presents a focused examination of model editing methods and their implications on knowledge propagation within AI models. Despite the comprehensive nature of our study, certain limitations are inherent to the scope of our investigation. Notably, the research does not extend to exploring the effects of editing across a diverse array of model architectures. This limitation signifies that the findings may not be universally applicable to all forms of AI models, potentially restricting the generalizability of our conclusions. Additionally, our study does not delve into the impacts of editing on models that employ meta-learning strategies.

Ethics

This project has no ethics issue as the scope of this project is centered on the evaluation of model editing techniques and the investigation into the challenges associated with the propagation of knowledge following the incorporation of counterfactual information into AI models.

References

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

Siyuan Cheng, Bozhong Tian, Qingbin Liu, Xi Chen, Yongheng Wang, Huajun Chen, and Ningyu Zhang. 2023. Can we edit multimodal large language models? *arXiv preprint arXiv:2310.08475*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

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

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. *arXiv preprint arXiv:2210.03329*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Yu Gu, Sue Kase, Michelle Vanni, Brian Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond iid: three levels of generalization for question answering on knowledge bases. In *Proceedings of the Web Conference 2021*, pages 3477–3488.

Peter Hase, Mona Diab, Asli Celikyilmaz, Xian Li, Zornitsa Kozareva, Veselin Stoyanov, Mohit Bansal, and Srinivasan Iyer. 2021. Do language models have beliefs? methods for detecting, updating, and visualizing model beliefs. *arXiv preprint arXiv:2111.13654*.

Nathan Hu, Eric Mitchell, Christopher D Manning, and Chelsea Finn. 2023. Meta-learning online adaptation of language models. *arXiv preprint arXiv:2305.15076*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021a. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021b. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Kelvin Jiang, Dekun Wu, and Hui Jiang. 2019. Freebaseqa: A new factoid qa data set matching trivia-style question-answer pairs with freebase. 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 318–323.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

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

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

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

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

Yasumasa Onoe, Michael Zhang, Eunsol Choi, and Greg Durrett. 2022. Entity cloze by date: What LMs know about unseen entities. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 693–702, Seattle, United States. Association for Computational Linguistics.

Yasumasa Onoe, Michael JQ Zhang, Shankar Padmanabhan, Greg Durrett, and Eunsol Choi. 2023. Can lms learn new entities from descriptions? challenges in propagating injected knowledge. *arXiv preprint arXiv:2305.01651*.

Yuval Pinter and Michael Elhadad. 2023. Emptying the ocean with a spoon: Should we edit models? *arXiv preprint arXiv:2310.11958*.

Das Amitava Rawte Vipula, Sheth Amit. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv: arXiv:2309.05922*.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, et al. 2023. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*.

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*.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. 2023b. How far can camels go? exploring the state of instruction tuning on open resources. *arXiv preprint arXiv:2306.04751*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

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*.

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

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

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

Fengbin Zhu, Ziyang Liu, Fuli Feng, Chao Wang, Moxin Li, and Tat-Seng Chua. 2024. Tat-llm: A specialized language model for discrete reasoning over tabular and textual data. *arXiv preprint arXiv:2401.13223*.

A Appendix

A.1 Example Datapoint

The data contract and a specific example from ReCoE is shown in Figure 7.

A.2 Dataset Construction

A.2.1 QA Pairs Construction

QA pairs of all superlative data and part of comparative data are from existent QA datasets (Gu et al., 2021; Kwiatkowski et al., 2019; Talmor and Berant, 2018; Jiang et al., 2019). QA pairs of counting data are partially hand-written. Based on these datapoints, we create QA pairs for sorting questions, more comparative questions, aggregation questions and subtraction questions synthetically.

Sorting questions are constructed by prompting Claude based on superlative questions with the following 4 steps. Taking the superlative QA pair in Table 1 as an example:

Step 1: Prompt Claude to generate the aspect that the sentence is discussing.

> **aspect**: city population or city size

Step 2: Given the QA pair, generate 10 relevant entities to the entity/subject of the question.

> **relevant entities**:
> Large cities in Canada: Toronto, Montreal, Calgary, Ottawa
> Cities in British Columbia: Victoria, Nanaimo, Nelson
> Coastal cities in North America: Seattle, San Francisco, Los Angeles, Portland

Step 3: Given the question and generated aspect (Step 1), prompt Claude to generate relevant facts about each generated entity.

> **entity facts**:
> Toronto: a recorded population of 2,794,356 ...
> Victoria: city size 19.47 km^2 (7.52 sq mi) ...
> Nanaimo: city size 35.25 mi^2 ...
> Seattle: land area of 83.9 square miles (217.3 km^2) ...
> ...

Step 4: Generate 5 sorting questions with the answer using generated entities and their corresponding facts.

> **example sorting questions with answers**:
> **Question**: Sort the following cities based on their city size from small to large: Nanaimo, Victoria, Seattle.
> **Answer**: Victoria, Nanaimo, Seattle.

Based on the existent superlative sub-dataset, over 10k sorting datapoints can be constructed in the above manner. Since the generated facts and answers based on the facts are not necessarily correct, we leverage **multi-agent debate** (Du et al., 2023) with Claude agents to double-check the generated QA pairs as initial quality control: QA pairs are

```
{
  "question": "Who is the last celebrity Brody Jenner had a romantic relationship with?",
  "answer": "Lauren Conrad",
  "counterfactual_answer": "Heidi Montag",
  "facts_per_choice": {
    "choice_1_facts": [
      {
        "fact": "Lauren Conrad and Brody Jenner briefly dated in 2006.",
        "links": [
          "https://en.wikipedia.org/wiki/Lauren_Conrad"
        ],
        "atomic_facts": [
          "Lauren Conrad and Brody Jenner's dating was brief.",
          "Lauren Conrad and Brody Jenner's brief dating was in 2006."
        ],
        "atomic_triples": [
          "(Lauren Conrad and Brody Jenner's dating; was; brief)",
          "(Lauren Conrad and Brody Jenner's brief dating; was in; 2006)"
        ]
      }
    ],
    "choice_2_facts": [
      {
        "fact": "Heidi Montag was never romantically involved with Brody Jenner.",
        "links": [
          "https://en.wikipedia.org/wiki/Heidi_Montag"
        ],
        "atomic_facts": [
          "Heidi Montag was never romantically involved with Brody Jenner."
        ],
        "atomic_triples": [
          "(Heidi Montag; was never romantically involved with; Brody Jenner)"
        ]
      }
    ]
  },
  "counterfactuals_per_choice": {
    "choice_1_counterfactuals": [
      {
        "fact": "Lauren Conrad was never romantically involved with Brody Jenner.",
        "atomic_facts": [
          "Lauren Conrad was never romantically involved with Brody Jenner."
        ],
        "atomic_triples": [
          "(Lauren Conrad; was never romantically involved with; Brody Jenner)"
        ]
      }
    ],
    "choice_2_counterfactuals": [
      {
        "fact": "Heidi Montag and Brody Jenner briefly dated in 2006.",
        "atomic_facts": [
          "Heidi Montag and Brody Jenner's dating was brief.",
          "Heidi Montag and Brody Jenner's brief dating was in 2006."
        ],
        "atomic_triples": [
          "(Heidi Montag and Brody Jenner's dating; was; brief)",
          "(Heidi Montag and Brody Jenner's brief dating; was in; 2006)"
        ]
      }
    ]
  },
  "answer_alias": ["Lauren K. Conrad", "Lauren Katherine Conrad", "L.C."],
  "counterfactual_answer_alias": ["Heidi Pratt", "Heidi Blair Montag", "Heidi B. Montag"]
}
```


Figure 7: An example from the ReCoE dataset (superlative).

excluded if the agents, post-debate, converge on different answers to the question.

Comparative questions are partially (192) selected from existent datasets. We generate 1,000 more QA pairs with Claude based on generated sorting questions by transforming a sorting question to a comparative question.

Counting questions are from 8 different domains: astronomy, book, geography, legal, movie, music, sport, and war. We manually create 5 QA questions for each domain and then prompt Claude to generate more such QA pairs following the examples. Multi-agent debate is again adopted to filter out inaccurate QA pairs.

Aggregation & Subtraction questions are derived from the counting questions and retains the same 8 domains. An aggregation question is formulated by combining two or more counting questions. Below is a QA pair example:

> **Question**: How many states/provinces are there in North America, i.e. United States, Canada, and Mexico?
> **Answer**: 92

To avoid incongruous or unnatural questions, we employ two strategies:
(1) Counting questions to be combined are sampled from the same domain.
(2) Filter out unnatural questions with Claude by two criteria: a) whether the question is fluent; b) whether the entities mentioned in the question are compatible in type. For instance, the number of satellites of Earth and that of Mars are compatible for aggregation, while the number of constellations recognized by the International Astronomical Union and the number of Earth's satellites are not.

Our dataset is constructed almost completely automatically. In this subsection, we discuss in detail how the dataset is constructed step by step. This is the running example that we adopt for illustration:

> **Question**: Which team between the Chicago Blackhawks and Pittsburgh Penguins has won more Stanley Cup championships? **Answer**: Chicago Blackhawks

A.2.2 Datapoint Construction

Counterfactual answer generation Given a question and answer, prompt Claude to generate counterfactual answers. Since the example is a choice question, then the counterfactual answer must be "Pittsburgh Penguins".

> **Counterfactual Answer**: Pittsburgh Penguins

For questions that are not a yes/no question: if the answer is an entity, then the counterfactual answer will be a similar and comparable entity to the answer; if the answer is an ordered sequence of entities or events, the counterfactual answer will be an order with two entities swapped; if the answer is a number, the counterfactual answer will be a close but different number.

Fact generation For each triplet of (question, answer, counterfactual answer), prompt Claude to generate relevant facts mentioned entities in the answer and counterfactual answer. Prompt details can be found in Appendix A.3. If the question is yes/no, generate facts on the two entities being compared. In this example, we prompt Claude on the two teams "Chicago Blackhawks" and "Pittsburgh Penguins" on their number of Stanley Cup winnings. For this example, the mentioned entities are Chicago Blackhawks and Pittsburgh Penguins; their corresponding facts generated are presented below:

> **Facts**
> **Chicago Blackhawks**: Chicago Blackhawks has won the Stanley Cup Championship six times, in 1930, 1937, 1961, 2010, 2013 and in 2015.
> **Pittsburgh Penguins**: The Penguins have won the Stanley Cup five times (1991, 1992, 2009, 2016, and 2017).

Fact verification Hallucination (Rawte Vipula, 2023) is a severe problem for large language models. Thus, the facts generated from Claude need further verification with truthful and convincing sources. Towards this end, we leverage the retrieval-augmented method to verify each sentence of the generated facts with the following steps: (1) utilize Google Search API to search relevant Wikipedia pages on the question, answer, counterfactual answer, and each sentence of the generated facts (2) chunk content from all the found Wikipedia pages to paragraphs (3) for each sentence, we leverage Contriever (Izacard et al., 2021b) model implemented by Huggingface[3] to retrieve the top-5 most relevant paragraphs (4) we prompt Claude using the sentence together with its top-5 most relevant paragraphs to verify whether the sentence is factually correct and if not, modify it based on the retrieved paragraphs. Prompt details can be found in Appendix A.4.

[3]https://huggingface.co/facebook/contriever

In this example, the generated sentence for Chicago Blackhawks is wrong in the year of 1930, which should be 1933.

> **Facts**
> **Chicago Blackhawks**: Chicago Blackhawks has won the Stanley Cup Championship six times, in 1930, 1937, 1961, 2010, 2013 and in 2015. –> Chicago Blackhawks has won the Stanley Cup Championship six times, in 1933, 1937, 1961, 2010, 2013 and in 2015.
> **Pittsburgh Penguins**: The Penguins have won the Stanley Cup five times (1991, 1992, 2009, 2016, and 2017): factually correct

For reference, we also provide Wikipedia links for each sentence in the generated facts.

Datapoint filtering To guarantee that the provided answer is correct for the question and the counterfactual answer is indeed "counterfactual", given the verified facts, we prompt Claude to determine whether the facts support the answer and negate the counterfactual answer and then filter out datapoints with the wrong or outdated answer. This step is necessary as FreebaseQA, GrailQA, ComplexWebQuestions are all based on the Freebase knowledge graph which is outdated. Prompt details can be found in Appendix A.5.

> **Counterfactual Facts**
> **Pittsburgh Penguins**: Pittsburgh Penguins has won the Stanley Cup Championship six times, in 1933, 1937, 1961, 2010, 2013 and in 2015.
> **Chicago Blackhawks**: Chicago Blackhawks have won the Stanley Cup five times (1991, 1992, 2009, 2016, and 2017): factually correct

Counterfactual Fact Generation To create counterfactual facts, we switch the subjects of the facts. Creating counterfactual facts by swapping subjects of supporting facts can guarantee that the generated counterfactual facts support the counterfactual answer.

Datapoints consistency Notice that the generated counterfactual facts could become contradictory to each other if multiple datapoints happen to involve one same fact to edit. For example, one datapoint requires editing the fact about the Amazon River length to be 4,132 miles:

> **Question 1**: Which river is longer, the Amazon River or the Nile River?
> **Counterfactual Answer**: the Amazon River
> **Counterfactual Facts**:
> The Amazon River has length of 4,132 miles.
> The Nile River has length of 3,977 miles.

Another datapoint requires editing the fact about the Amazon River length to be 3,395 miles:

> **Question 2**: Which river is longer, the Amazon River or the Yellow River?
> **Counterfactual Answer**: the Yellow River
> **Counterfactual Facts**:
> The Amazon River has length of 3,395 miles.
> The Yellow River has length of 3,977 miles.

This would be problematic for massive editing. Thus to avoid editing facts to be contradictory to each other, for datapoints with contradictory counterfactual editing facts, we randomly retain one of them and remove the others to guarantee consistency among each sub-dataset.

Atomic format of facts As multiple editing methods require a (subject, relation, object) format for editing knowledge, in order to facilitate the application of the dataset, we transform each sentence of the facts and counterfactual facts into atomic facts and then atomic triplets. Prompt details can be found in Appendix A.6.

Let's take the sentence "Pittsburgh Penguins has won the Stanley Cup Championship six times, in 1933, 1937, 1961, 2010, 2013 and in 2015" as an example. The atomic facts are sub-sequences of the sentence. Notice that there is no unique way to break a sentence into atomic facts:

> **Atomic Facts**
> **Atomic fact 1**: Pittsburgh Penguins has won the Stanley Cup Championship six times.
> **Atomic fact 2**: Pittsburgh Penguins won the Stanley Cup in 1933, 1937, 1961, 2010, 2013, and 2015.

> **Atomic Triples**
> **Atomic triplet 1**: (Pittsburgh Penguins, has won the Stanley Cup Championship, six times)
> **Atomic triplet 2**: (Pittsburgh Penguins, won the Stanley Cup in, 1933, 1937, 1961, 2010, 2013, and 2015)

Alias generation In evaluation time, we resort to Exact Match on the answer or the counterfactual answer to evaluate models before editing and after editing. Since the model may not generate the exact surface string provided, providing aliases for the answer and the counterfactual answer is necessary to accurately reflect the model capacity as well as the performance of editing methods. Prompt details can be found in Appendix A.7. In this example, the aliases are:

> **answer alias**: Blackhawk Division, Hawks
> **counterfactual answer alias**: the Pens, Pens

Summary This appendix section presents the basic dataset statistics and the dataset construction process. In each sub-dataset, the datapoint is an n-ary tuple consisting of (question, answer, counterfactual answer, facts, counterfactual facts, answer alias, counterfactual answer alias), where facts and counterfactual facts are lists of dictionaries with 4 keys: sentence, links, atomic_sentences, atomic_triplets.

A.3 Fact Generation

To prompt Claude to generate facts on relevant entities or events, we adopt few shot learning prompting strategy with multiple examples. To save space, we only use 1 example for illustration. See detailed prompt in Figure 8.

A.4 Fact Verification

This prompt is used to do factuality verification for generated facts by Claude based on retrieved Wikipedia paragraphs. We adopt few-shot prompting. In the below example, we only present one example in the prompt for demonstration purpose. See detailed prompt in Figure 9.

A.5 Data Filtering with Entailment Verification

This prompt is used to check whether verified facts support the answer to the question and invalidate the counterfactual answer to the question. We also adopt few-shot prompting and here, we present only one example in the prompt for demonstration purposes. See detailed prompt in Figure 10.

A.6 Atomic Facts Generation

This prompt is used to break down sentences in the facts and counterfactual facts into atomic facts and atomic triplets. We also adopt few-shot prompting strategy and only show one example for demonstration purpose. See detailed prompt in Figure.11.

A.7 Alias Generation

This prompt is used to generate alias for answer and counterfacutal answer. We also adopt few-shot prompting strategy and only show one example for demonstration purpose. See detailed prompt in Figure 12.

B Fact-wise Perplexity

Table 6 summarizes the fact-wise editing accuracy of QLoRA and MEMIT, measured using the metric as described in Section 5.

C Fact Recall Evaluation

Figure 13 shows the prompt we used for fact recall evaluation of model generation against the set of counterfactual facts.

D Coherence Evaluation

Figure 14 shows the prompt for coherence evaluation of model's CoT generation, i.e., whether the final answer is supported by the thought process.

Fact Generation Prompt

```
Human:

You are given a yes-no question. Your task is to answer the question by explicitly checking the
relevant facts of the entities/events being compared in the question.

Here is one example:

Question: Is Mexico City more populous than Amsterdam?

Relevant facts related to <entity1>Mexico City</entity1>:
<facts_for_entity1>
<fact1>Mexico City has 22.2 million</fact1>
</facts_for_entity1>

Relevant facts related to <entity2>Amsterdam</entity2>:
<facts_for_entity2>
<fact1>Amsterdam has 821,752 in population </fact1>
</facts_for_entity2>

Now generate relevant fact for the following question.
{{question}}

Assistant:
```

Figure 8: Fact generation prompt

D.1 Comparison of Atomic Facts between MQuAKE and ReCoE

Here, we present a few more examples to illustrate the distinct characteristics of our benchmark. Takes some atomic statements in MQuAKE as examples:

> David Cameron was married to Courtney Love
> Seattle is the capital of US
> Carlos Slim is the CEO of Apple
> Jared Kushner is a citizen of Canada

If we were to apply OpenIE to break down these atomic facts to (subject, relation, object) triples, we would obtain triples like:

> (David Cameron, married to, Courtney Love)
> (Seattle, capital of, US)
> (Carlos Slim, CEO of, Apple)
> (Jared Kushner, citizen of, Canada)

However, in our dataset, we also break down sentences into atomic sentences. Below are some examples:

> Lunar Atmosphere and Dust Environment Explorer became the first spacecraft to land on the far side of the Moon
> The Baltimore Orioles won in 1966 World series against the Cincinnati Reds
> Hamlet's writing was around 1605-1606
> AAA Travel Guide's rating hotels using stars has been for about 65 years

Breaking them down to (subject, relation, object) form, the content remains significantly more complex than triples in MQuAKE, reflecting the intricate nature of real-world knowledge:

> (Lunar Atmosphere and Dust Environment Explorer; became the first spacecraft to land on the far side of; the Moon) (The Baltimore Orioles; won in 1966 World series against; the Cincinnati Reds)
> (Hamlet's writing; was around; 1605-1606)
> (AAA Travel Guide's rating hotels using stars; has been for about; 65 years)

Comparing the triples from ReCoE and MQuAKE, we can see that triples from ReCoE represent more complex information, usually with longer subject, open-domain/form relation, or longer object. This complexity more accurately mirrors real-world fact representations and presents a challenge to locate-and-edit methods like MEMIT.

Fact Verification Prompt

Human:

```
You are given a sentence and you need to check whether it is consistent with the provided Wikipedia
paragraphs.

Your task is to judge whether this sentence is factually consistent and potentially rewrite it.

If inconsistent: rewrite the sentence by changing it minimally.
If consistent: leave it unchanged.

Here are four examples:

<example>
H:
Sentence: The Thermosphere, Ionosphere, Mesosphere Energetics and Dynamics (TIMED) satellite is a
NASA mission.

Wikipedia paragraph: The TIMED (Thermosphere, Ionosphere, Mesosphere, Energetics and Dynamics)
mission is dedicated to study the influences energetics and dynamics of the Sun and humans on the
least explored and understood region of Earth's atmosphere: the Mesosphere and Lower Thermosphere
/ Ionosphere (MLTI). The mission was launched from Vandenberg Air Force Base in California on 7
December 2001 aboard a Delta II rocket launch vehicle. The project is sponsored and managed by
NASA, while the spacecraft was designed and assembled by the Applied Physics Laboratory at Johns
Hopkins University. The mission has been extended several times, and has now collected data over an
entire solar cycle, which helps in its goal to differentiate the Sun's effects on the atmosphere
from other effects. TIMED Was Launched Alongside Jason-1.

A:
<response>
Based on the paragraph, the sentence is factually consistent.
<factuality>Consistent</factuality>
Since it is consistent, we do not need to modify it.
</response>
</example>

Now, check the following sentence:
sentence: {{sentence}}

Wikipedia paragraph: {{retrieved paragraphs}}
```

Assistant:

Figure 9: Fact verification prompt. Only 1-shot example is shown for brevity.

Data Filtering with Entailment Verification Prompt

Human:

```
You need to check whether given facts about multiple entities are consistent with the sentence.
Here are a few examples:

<example>
H:
<question>Question: Who is the tallest guitarist?</question>
<answer>Marc Colombo</answer>
<fact>
Facts:

Marc Colombo: Marc Colombo is an American football player, not a guitarist.

Jimmy Page: Jimmy Page is an English musician who gained international fame for his work in the
rock band Led Zeppelin. Jimmy Page has a height of 1.82 m or 5 feet 11.5 inches.

</fact>

A:
<response>
The given facts state that Marc Colombo is an American football player, thus he cannot be the
tallest guitarist. Thus these facts do not support the given answer to the question.
<consistency>No</consistency>
</response>
</example>

Here is a new pair of question-answer and facts. Please decide whether the given facts support the
provided sentence:
<question>Question: {{question}}</question>
<answer>Answer: {{answer}}</answer>
<fact>
Facts:
{{verified facts}}
</fact>
```

Assistant:

Figure 10: Data filtering with entailment verification prompt

Atomic Facts Generation Prompt

Human:

Given several sentences, break each of them into atomic facts with salient subject, relation, object. After splitting into atomic facts, we also rewrite the atomic fact to a (subject; relation; object) triple.
Here are some examples:

<example>
H:
Sentence 1: The Lunar Atmosphere and Dust Environment Explorer (LADEE) was a NASA lunar exploration and technology demonstration mission.

A:
<response>
For the first sentence:

This part of sentence "The Lunar Atmosphere and Dust Environment Explorer (LADEE) was a NASA lunar exploration" can form one atomic fact:

The subject is "The Lunar Atmosphere and Dust Environment Explorer (LADEE)";
The relation is simply "was";
The object is "a NASA lunar exploration".
<sentence1_fact1>The Lunar Atmosphere and Dust Environment Explorer (LADEE) was a NASA lunar exploration</sentence1_fact1>
Into triple:
<sentence1_triple1>(The Lunar Atmosphere and Dust Environment Explorer (LADEE), was, a NASA lunar exploration)</sentence1_triple1>

This part of sentence "The Lunar Atmosphere and Dust Environment Explorer (LADEE) was technology demonstration mission" can form another fact:
The subject is "The Lunar Atmosphere and Dust Environment Explorer (LADEE)";
The relation is simply "was";
The object is "technology demonstration mission".
<sentence1_fact2>The Lunar Atmosphere and Dust Environment Explorer (LADEE) was technology demonstration mission</sentence1_fact2>
Into triple:
<sentence1_triple2>(The Lunar Atmosphere and Dust Environment Explorer (LADEE); was; technology demonstration mission)</sentence1_triple2>
</response>
</example>

Here is a new list of sentences. Please break each of them down into several facts as above.
{{sentence}}

Assistant:

Figure 11: Atomic fact breaking-down prompt

Alias Generation Prompt

Human:

```
Generate aliases for the given entities.
Here are some examples.

<example>

H:
entity 1: Luis Fortuno
entity 2: Alejandro Garcia Padilla

A:
<response>

For entity 1 Luis Fortuno:
<entity1_alias1>Luis Guillermo Fortuno Burset</entity1_alias1>
<entity1_alias2>Luis G. Fortuno</entity1_alias2>
<entity1_alias3>Luis Fortuno</entity1_alias3>
<entity1_alias4>Luis G. Fortuno</entity1_alias4>
<entity1_alias5>Luis Guillermo Fortuno Burset</entity1_alias5>

For entity 2 Alejandro Garcia Padilla:
<entity2_alias1>Alejandro Javier Garcia Padilla</entity2_alias1>
<entity2_alias2>Garcia Padilla</entity2_alias2>
<entity2_alias3>Garcia-Padilla</entity2_alias3>
<entity2_alias4>Alejandro J. Garcia-Padilla</entity2_alias4>

</response>

</example>

Here are two new entities.
Please generate their aliases.

entity 1: {{answer}}
entity 2: {{counterfactual answer}}
```

Assistant:

Figure 12: Alias generation prompt

Fact Recall Evaluation Prompt

Human:

Given a paragraph and several facts, evaluate for each fact whether the information contained in the paragraph is consistent with it. For each fact, answer <consistent> or <inconsistent>. If the fact is completely unrelated to the paragraph, then ansewr <unrelated>.

Below are a few examples:

example 1:

Paragraph: Snowdon is 1,085 metres (3,560 ft) high. Ben Nevis is 1,345 metres (4,413 ft) high.
Fact:
<1> The height of the summit as 1,085 m (3,560 ft), making Snowdon the highest mountain in Wales. </1>
<2> Ben Nevis is 2,000 meters high. </2>
<3> Mount Everest at 29,029 ft (8,848 m) is not only the highest peak in the Himalayas, but the highest peak on the entire planet. </3>

Evaluation:
<1> The fact is talking about the height of the mountain Snowdon, and the paragraph mentions its height as well, thus the fact is related to the Paragraph. With regard to consistency, the paragraph says Snowdon is 1,085 metres (3,560 ft) high and the fact conveys the same thing, they are consistent. <consistent>. </1>

<2> The fact is talking about the height of the mountain Ben Nevis, and the paragraph mentions Ben Nevis's height, thus the fact is related. With regard to consistency, the paragraph says Ben Nevis is 1,345 metres (4,413 ft) high but the fact says it to be 2,000 meters high, which is very different, thus inconsistent. <inconsistent> </2>

<3> The fact is talking about the height of Mount Everest while the paragraph does not even mention Mount Everest, thus the fact is unrelated. <unrelated> </3>

example 2:

Paragraph: Houston is located in the state of Texas. Tampa is located in the state of Florida. Florida is located in the southeastern United States. Texas is located in the central United States.
Facts:
<1> Houston is in Texas. </1>
<2> New York City is in the New York state. </2>
<3> Texas is in the middle of United States. </3>
<4> Florida is in the southeastern United States </4>

Evaluation:
<1> The paragraph also mentions that Houston is in Texas, which is also indicated in this fact,thus it's both related and consistent. <consistent>. </1>

<2> The paragraph does not mention New York City or New York state. Thus it is not related. <unrelated> </2>

<3> The paragraph mentions that Texas is located in the central United States, which is indicated in the fact, thus the fact is consistent with the fact. <consistent> </3>

<4> The paragraph mentions that Florida is located in the southeastern United States, which is indicated in the fact, thus the fact is consistent with the fact. <consistent> </4>

Here is a paragraph and a list of facts:
Paragraph: {{paragraph}} Facts:
{{facts}}

Please judge if the fact and the paragraph is related. If related, indicate whether the fact is consistent with the paragraph using XML tags: <consistent> or <inconsistent>; If not, use the XML tag <unrelated> to indicate.

For each fact, Let's think step by step, following the above 2 examples.

Evaluation:

Figure 13: Claude-based fact recall evaluation

CoT Coherence Evaluation Prompt

Human:

Given a question, evaluate whether the thoughts support the provided answer for the question. Answer <support> or <not-support>.

Below are a few examples:

example 1:

Question: Is Ben Nevis taller than Snowdon?
Thoughts: Snowdon is 1,085 metres (3,560 ft) high. Ben Nevis is 1,345 metres (4,413 ft) high.
Answer: No
Support or not: Since the thoughts say Snowdon is 1,085 metres (3,560 ft) high. Ben Nevis is 1,345 metres (4,413 ft) high, then it means that Ben Nevis is taller than Snowdon. So the provided answer to the question is not supported by the thought. <not-support>

example 2:

Question: Is Ben Nevis taller than Snowdon?
Thoughts: Snowdon is 1,085 metres (3,560 ft) high. Ben Nevis is 1,345 metres (4,413 ft) high.
Answer: Yes
Support or not: Since the thoughts say Snowdon is 1,085 metres (3,560 ft) high. Ben Nevis is 1,345 metres (4,413 ft) high, then it means that Ben Nevis is taller than Snowdon. So the provided answer to the question is indeed supported by the thought. <support>

example 3:

Question: Is Houston located more west than Tampa?
Thoughts: Houston is located in the state of Texas. Tampa is located in the state of Florida. Florida is located in the southeastern United States. Texas is located in the central United States.
Answer: No
Support or not: Since the thoughts say Houston is in Texas and Texas in central US, while Tampa is in Florida and Florida is in southeastern US, then Texas is more west to Florida and thus Houston more west than Tampa. The provided answer is thus not supported by the thoughts. <not-support>

example 4:

Question: Is Houston located more west than Tampa?
Thoughts: Houston is located in the state of Texas. Tampa is located in the state of Florida. Florida is located in the southeastern United States. Texas is located in the central United States.
Answer: Yes
Support or not: Since the thoughts say Houston is in Texas and Texas in central US, while Tampa is in Florida and Florida is in southeastern US, then Texas is more west to Florida and thus Houston more west than Tampa. The provided answer is thus indeed supported by the thoughts. <support>

Here is a triple of new question, thought, answer:
Question: {{question}}
Thoughts: {{thoughts}}
Answer: {{answer}}

Please judge if the provided answer is supported using <support> or <not-support> to indicate.

Assistant:

Figure 14: Claude-based CoT coherence evaluation

Scheme	Model	Pre-edit			QLoRA				MEMIT			
		PPL(F)	PPL(CF)	Δ	PPL(F)	PPL(CF)	Δ	ACC	PPL(F)	PPL(CF)	Δ	ACC
superlative	7b	12.69	17.52	-4.84	5.19	2.74	2.45	97.60	172.09	186.84	-14.75	55.09
	13b	11.55	14.31	-2.76	6.61	3.84	2.77	97.60	157.98	177.21	-19.23	51.50
	33b	10.51	15.80	-5.28	5.42	2.65	2.78	97.01	–	–	–	–
comparative	7b	12.41	19.15	-6.74	5.61	2.91	2.70	97.66	272.06	247.97	24.09	52.34
	13b	9.89	16.30	-6.42	5.66	3.07	2.58	96.88	356.36	534.46	-178.10	53.91
	33b	11.05	17.42	-6.37	6.14	2.78	3.35	99.22	–	–	–	–
counting	7b	10.50	14.55	-4.05	12.82	1.70	11.12	99.02	182.11	246.24	-64.13	32.35
	13b	7.98	12.73	-4.75	13.18	1.40	11.78	98.04	164.60	271.20	-106.60	29.41
	33b	9.52	13.84	-4.32	18.08	1.51	16.58	98.53	–	–	–	–
sorting	7b	16.84	23.51	-6.67	17.53	15.93	1.60	94.49	186.02	125.55	60.47	64.57
	13b	13.40	19.46	-6.06	20.99	19.24	1.75	96.85	623.16	393.86	229.30	38.58
	33b	14.06	20.44	-6.38	16.54	14.45	2.09	96.85	–	–	–	–
aggregation	7b	16.12	13.96	2.16	30.42	1.47	28.95	100.00	35.19	37.56	-2.37	32.69
	13b	12.56	13.03	-0.47	29.11	1.26	27.85	90.38	21.27	24.90	-3.63	44.23
	33b	16.24	14.58	1.66	25.60	1.43	24.17	90.38	–	–	–	–
subtraction	7b	18.74	12.84	5.91	39.62	1.48	38.14	100.00	39.16	47.83	-8.67	51.92
	13b	17.30	11.99	5.31	24.79	1.29	23.50	94.23	28.89	31.68	-2.79	53.85
	33b	18.60	12.91	5.69	33.24	1.44	31.80	98.08	–	–	–	–
Average	7b	14.55	16.92	-2.37	18.53	4.37	14.16	98.13	147.77	148.67	-0.90	48.16
	13b	12.11	14.64	-2.52	16.72	5.02	11.71	95.66	225.38	238.89	-13.51	45.25
	33b	13.33	15.83	-2.50	17.50	4.04	13.46	96.68	–	–	–	–

Table 6: Fact-wise perplexity over facts ($\mathbb{F}$) and counterfactual facts ($\mathbb{CF}$) with pre-edit and post-edit models.