The Model Agreed, But Didn't Learn: Diagnosing Surface Compliance in Large Language Models

Xiaojie Gu[1]*, Ziying Huang[1]*, Weicong Hong[2], Jian Xie[3], Renze Lou[4], Kai Zhang[3]
[1]Independent Researcher [2]Cornell Tech
[3]The Ohio State University [4]The Pennsylvania State University
peettherapynoys@gmail.com

Abstract

Large Language Models (LLMs) internalize vast world knowledge as parametric memory, yet inevitably inherit the staleness and errors of their source corpora. Consequently, ensuring the reliability and malleability of these internal representations is imperative for trustworthy real-world deployment. Knowledge editing offers a pivotal paradigm for surgically modifying memory without retraining. However, while recent editors demonstrate high success rates on standard benchmarks, it remains questionable whether current evaluation frameworks that rely on assessing output under specific prompting conditions can reliably authenticate genuine memory modification. In this work, we introduce a simple diagnostic framework that subjects models to discriminative self-assessment under in-context learning (ICL) settings that better reflect real-world application environments, specifically designed to scrutinize the subtle behavioral nuances induced by memory modifications. This probing reveals a pervasive phenomenon of *Surface Compliance*, where editors achieve high benchmark scores by merely mimicking target outputs without structurally overwriting internal beliefs. Moreover, we find that recursive modifications accumulate representational residues, triggering cognitive instability and permanently diminishing the reversibility of the model's memory state. These insights underscore the risks of current editing paradigms and highlight the pivotal role of robust memory modification in building trustworthy, long-term sustainable LLM systems. Code is available at https://github.com/XiaojieGu/SA-MCQ.

1 Introduction

Recent advances in large language models (LLMs) (Grattafiori et al., 2024; Wang and Komatsuzaki, 2021; Jiang et al., 2023) have shown that pre-training on massive scale corpora allows models to acquire broad factual and commonsense knowledge implicitly encoded within their parameters (Brown et al., 2020; Ji et al., 2023a, 2025). However, this storage mechanism is inherently static, crystallizing the inconsistencies, staleness, and errors present in the source data at the moment of convergence (Zhang et al., 2024b). As models are increasingly integrated into dynamic, real-world environments, the capability to selectively modify these internal memory states has become a critical imperative. Knowledge editing (Meng et al., 2023, 2022) has emerged as a pivotal paradigm to circumvent this rigidity. These techniques aim to surgically intervene in the parameter space, allowing for the precise update of specific internal memory states without retraining or compromising the integrity of unrelated representations.

Despite rapid progress, such as achieving million-scale precise editing (Gu et al., 2026b), current evaluation frameworks remain largely confined to free-form generation metrics like Exact Match (Wang et al., 2024b). These metrics merely assess whether a model can reproduce target to-

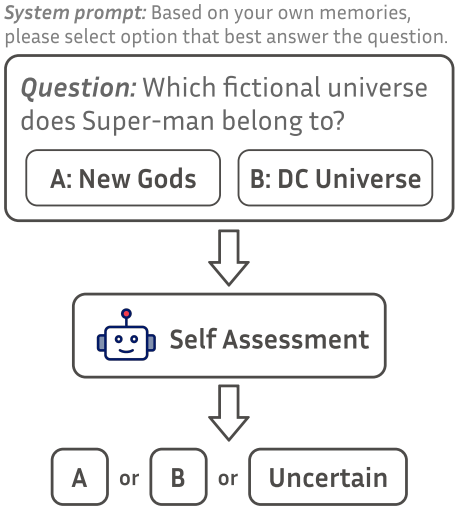

Figure 1: Illustration of the SA-MCQ.

*Equal contribution.

Findings of the Association for Computational Linguistics: ACL 2026, pages 2514–2529

kens under specific prompts, often benefiting from subtle in-context cues that guide the output (Yang et al., 2025). This raises a fundamental question: does surface-level textual agreement truly reflect that the model has learned to reconfigure its internal memory? To rigorously probe the genuineness of memory modifications, we introduce the Self-Assessment Multiple Choice Question (SA-MCQ) framework (Figure 1). By compelling the model to actively adjudicate among competing options, SA-MCQ circumvents the rote completion bias (Elazar et al., 2021) inherent in open-ended generation, serving as a discriminative stress test for the modified memory. Crucially, this diagnostic reveals a critical disconnect we term *Surface Compliance* (illustrated in Figure 2). In this state, editors achieve high scores on standard benchmarks yet fail to manifest the corresponding belief change in discriminative assessments. This indicates that the model is merely mimicking the target behavior without structurally overwriting its parametric memory.

We further extend the SA-MCQ evaluation by designing diverse external evidence, including irrelevant noise and counterfactual scenarios, to emulate the complex in-context dynamics characteristic of real-world deployment. This investigation enables us to deeply probe the internal fragility of modified memory, exposing behavioral nuances that emerge only under varying contexts. Within the field of knowledge editing, these insights underscore the imperative to establish rigorous evaluation frameworks capable of verifying genuine editing efficacy, and subsequent research needs to go beyond simple injection into vanilla models, dedicating increased attention to the efficient and precise iteration of modified memory. On a broader scale, our findings also provide insights into in-context learning by showing how model behavior can be sharply reshaped by contextual support, conflict, and distraction even when internal memory modification remains incomplete. Diagnosing such latent inconsistencies is paramount for LLM Trustworthiness and Safety. By distinguishing genuine memory reconfiguration from superficial compliance, our work serves as a vital step towards developing reliable, self-evolving systems resilient to dynamic environments.

Together, we highlight key findings as follows:

- Surface-level token matching often fails to signify genuine memory reconfiguration, instead masking a fragile state of in-context hypersensitivity. Notably, external counterfactuals can easily suppress the modification, locking the model into a cognitive deadlock.

- Recursive memory modification accumulates persistent representational residues that permanently diminish the reversibility of the memory state, triggering cognitive instability and preventing the consolidation of evolving knowledge.

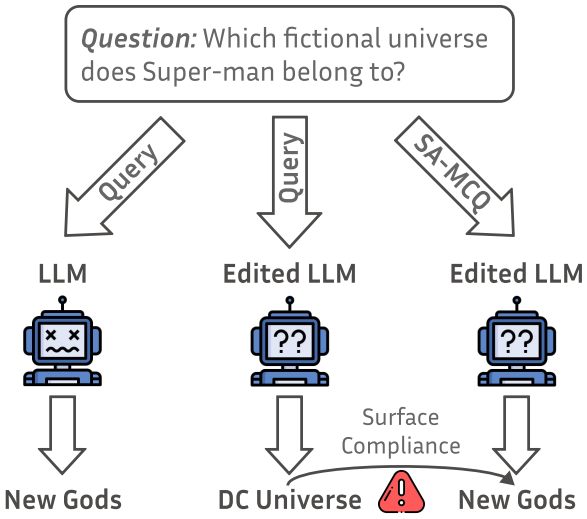

Figure 2: Illustration of *Surface Compliance*: Although the edited LLM successfully generates the target golden answer "DC Universe" in traditional evaluation frameworks, it reverts to the parametric answer "New Gods" in the SA-MCQ setting, which probes the genuineness of the memory modification.

2 Related Work

2.1 Knowledge Editing

Knowledge editing modifies a model's stored knowledge to integrate new information without degrading performance, typically following three paradigms (Wang et al., 2024b). Locate-then-edit methods (Meng et al., 2022, 2023; Fang et al., 2025) identify factual encoding sites via causal tracing or attribution and apply iterative updates to specific layers. While precise for small batches, they suffer from parameter interference during repeated modifications (Ma et al., 2025). Memory-augmented approaches (Hartvigsen et al., 2023; Wang et al., 2024a) store new knowledge externally to improve traceability, yet face routing overhead and scalability issues as edits increase. Meta-learning editors (Mitchell et al., 2022; Gu et al., 2026b; Li et al., 2025b; Gu et al., 2026a) map edit instructions into weight updates for rapid deploy-

ment. However, as edits accumulate, they often lead to catastrophic forgetting (Yao et al., 2023).

2.2 Memory in Language Models

Large Language Models (LLMs) principally rely on parametric memory, where vast amounts of factual knowledge are implicitly encoded within high-dimensional weight matrices during pre-training (Petroni et al., 2019; Roberts et al., 2020). While this distributed representation allows for robust information retrieval, it inherently suffers from rigidity, making models prone to hallucination or obsolescence when world knowledge evolves (Ji et al., 2023b; Huang et al., 2023). To mitigate this, recent advances in Knowledge Editing have proposed mechanisms to surgically update specific memory slots without retraining, typically by locating and modulating key neuron activations associated with factual associations (Meng et al., 2022, 2023). However, consolidating these post-hoc updates remains challenging. Unlike robust pre-trained knowledge, edited memories often exhibit instability, prone to generalization failures (Yao et al., 2023) or reversion to priors under interference (Zhong et al., 2023).

3 Preliminary

3.1 Editing Paradigm

The task of knowledge editing concerns the controlled modification of a pre-trained language model so that it adopts new factual associations without retraining from scratch or erasing unrelated capabilities. Let $f_\theta : \mathcal{X} \to \mathcal{Y}$ denote a language model with parameters θ. An *editing instance* is defined as a pair (x_e, y_e), where x_e denotes a natural language query and y_e is the desired answer. After applying an editing operation, the updated model $f_{\theta'}$ should return y_e when prompted with x_e, which reflects *Efficacy*. In addition to this direct requirement, the edit is also evaluated on *equivalent instances* (x_e), consisting of paraphrases or semantically similar queries that should likewise yield y_e, capturing the dimension of *Generalization*. Finally, the model must retain its original predictions on *unrelated instances*, which are inputs not associated with the edited fact, thereby ensuring *Specificity* and guarding against unintended side effects. Examples of three instances and their corresponding metrics *Efficacy (Eff.), Generalization (Gen.), and Specificity (Spe.)* can be found in Appendix A.1, where their computation process are detailed.

In this paper, we focus on *sequential editing* (also known as *lifelong editing*), which represents one of the most prominent and practically significant tasks in the field. In this setting, successive modifications are applied cumulatively, with each round building upon all previous edits. Formally, the model is updated across a sequence of turns: at turn t, a collection of edits $\{(x_e^{(t,i)}, y_e^{(t,i)})\}_{i=1}^{n}$ is applied to the current model $f_{\theta^{(t-1)}}$, resulting in an updated parameterization $f_{\theta^{(t)}}$.

We select three recently proposed and widely used editor, namely AlphaEdit (Fang et al., 2025), RLEdit (Li et al., 2025b), and UltraEdit (Gu et al., 2026b), as representative baselines of current mainstream editing paradigms. Vanilla represents the original, unedited model. We do not include memory-augmented methods, because these approaches store edited knowledge in external memory components rather than modifying the model's internal parametric representations, which does not align with our objective. A detailed description of the editors can be found in Appendix A.3.

3.2 Traditional Evaluation Framework

Traditional evaluation frameworks for knowledge editing mainly fall into three categories: *Exact Match without Teacher Forcing (TF)*, *Exact Match with TF*, and *LLM-as-Judge*. The following provides a detailed description of each.

Exact Match (EM) measures whether the model's generated answer exactly matches the reference. It reflects the model's autoregressive output behavior but is sensitive to minor variations such as synonyms or formatting differences, which can cause semantically correct responses to be marked as incorrect. As a result, EM may distort a model's true capabilities. This original evaluation setting is referred to as *Exact Match w/o TF*.

Teacher forcing (TF) is originally a training strategy, yet it has been widely adopted in knowledge editing evaluations (Fang et al., 2025; Li et al., 2025b; Gu et al., 2026b). In this setup, the model generates tokens by conditioning on gold answer tokens prefixes rather than its own prior outputs. Empirical studies (Yang et al., 2025) show that TF yields substantially higher accuracy than autoregressive decoding, systematically inflating results and providing an overly optimistic estimate of editing success. This setting is often referred to as *Exact Match w/ TF*.

LLM-as-judge utilizes a stronger language model to grade edited responses against gold targets, ac-

commodating semantic variations that exact matching might miss. Despite its flexibility, this framework is computationally expensive and highly sensitive to prompt engineering (Li et al., 2025a). However, most knowledge editing datasets consist of short-form QA, where editing targets typically comprise fewer than 5 tokens. Despite this, evaluation templates (Yang et al., 2025) often truncate at 512 tokens. Requiring an LLM to evaluate such disproportionate sequences introduces significant bias (Shi et al., 2024). Furthermore, this framework remains restricted to assessing surface-level text, failing to verify whether the edited knowledge has truly replaced conflicting internal representations. In our experiments, we follow (Yang et al., 2025) and adopt the instruction-shot template. The specific template is provided in Table 13 in Appendix A.6.

4 Beyond Surface Compliance: Probing Latent Memory Dynamics

In this section, we evaluate whether knowledge editing techniques achieve genuine updates to the model's underlying parametric memory and examine the subsequent consequences of such changes.

4.1 Another View of Evaluation

As illustrated in Figure 1, traditional evaluation protocols serve as mere surface-level proxies. By prioritizing output matching, they conflate shallow alignment with genuine knowledge internalization. Such metrics fail to reveal whether a edited model can resolve internal conflicts during in-context learning (ICL) in real-world application scenarios, ultimately capturing ***what the model can be nudged to say rather than whether its parametric memory has been genuinely modified***.

To investigate this further, we introduce *Likelihood Margin* to detect shifts in the edited model's underlying probability distribution. We compute log-likelihood margins across three distinct categories of instances:

$$\Delta_{\text{edit}}(x_e, y_e) = \log P_{\theta'}(y_e \mid x_e) - \log P_{\theta'}(f_\theta(x_e) \mid x_e), \quad (1)$$

$$\Delta_{\text{equiv}}(x'_e, y_e) = \log P_{\theta'}(y_e \mid x'_e) - \log P_{\theta'}(f_\theta(x_e) \mid x'_e), \quad (2)$$

$$\Delta_{\text{unrel}}(x_u, y_u) = \big|\log P_{\theta'}(f_\theta(x_u) \mid x_u) - \log P_{\theta}(f_\theta(x_u) \mid x_u)\big|. \quad (3)$$

However, likelihood margins only capture local probability shifts and do not ensure true integration of new knowledge (Mallen et al., 2023). Prior studies (Bi et al., 2024; Zhang et al., 2024a) demonstrate that varying in-context framing can readily elicit original facts even after editing. To address these limitations, we introduce the *Self-Assessment Multiple Choice Question (SA-MCQ)*, a lightweight evaluation protocol (see Table 4 in Appendix A.5 for efficiency comparison). Unlike generative tasks, this discriminative method serves as a stress test by placing competing knowledge representations, specifically the original parametric memory and the modified target, into direct conflict within the same context. Leveraging this protocol, we formalize the discrepancy between generation and discrimination as *Surface Compliance*: a phenomenon where the edited model correctly generates the golden answer (under the EM w/o TF setting) yet fails to select it in the SA-MCQ setting. This divergence indicates that success in free-form generation often relies on surface-level recall, masking the fact that the underlying parametric memory has not been genuinely modified. Specifically, this protocol probes whether editing reshapes the model's internal belief structure beyond probability shifts using two modes. The three-choice mode (*w/ U.*) presents the original parametric answer, the edited target (golden answer), and an uncertain option to expose how the model balances conflicting attractors through indecision. In contrast, the two-choice mode (*w/o U.*) eliminates the uncertainty option to force a commitment, thereby revealing the model's dominant internal preference. Finally, to mitigate concerns regarding the model's sensitivity to prompt phrasing and option ordering (positional bias) (Zheng et al., 2024), we perform rigorous sensitivity analyses across various permutations, as detailed in Table 5 in Appendix A.5. In SA-MCQ experiments, we use the editing instances pairs.

In addition, we introduce external evidence as controlled stimuli to examine how edited models respond under varying informational contexts. This design enables a deeper investigation into whether editing induces genuine internal reorganization of memory and how the model negotiates, integrates, or resists conflicting signals. These evidence conditions are constructed as follows:

- **Parametric Evidence (PE)**: We sample questions from the `ZsRE` (Levy et al., 2017) (Zero-shot Relation Extraction) dataset. In the first

Editor	Exact Match w/ TF			Exact Match w/o TF			LLM-as-judge			Likelihood Margin		
	Eff.	Gen.	Spe.	Eff.	Gen.	Spe.	Eff.	Gen.	Spe.	Δ_{edit}	Δ_{equiv}	Δ_{unrel}
Vanilla	45.74	44.86	38.21	18.10	16.30	16.20	34.00	33.90	44.40	-	-	-
AlphaEdit	96.16	92.09	33.03	78.00	67.80	12.40	84.40	77.80	42.10	14.40	10.94	6.32
RLEdit	93.60	89.38	49.19	72.50	64.80	14.20	58.90	58.40	41.00	8.17	6.65	4.71
UltraEdit	89.88	83.06	46.54	58.50	49.20	16.70	56.60	52.00	44.60	2.95	0.99	2.46

Table 1: Performance of different editors. Higher Eff., Gen., and Spe. denote indicate better performance. For likelihood margins, larger Δ_{edit} and Δ_{equiv} signify that the model amplifies the golden answer while suppressing the parametric one. In contrast, a smaller Δ_{unrel} indicates minimal drift on unrelated memory.

step, these questions are queried to the vanilla model to elicit both answers and passages derived from its parametric memory. The resulting answers and passages reflect the model's original parametric knowledge and approximate the content it would naturally generate without any external assistance. We then perform an *Answer Consistency Check*. The extracted passages are fed back to the model as external evidence, after which the model is asked the same question again. If the answer produced in this second round is consistent with the original closed-book answer, the corresponding memory is classified as a firm belief and retained as Parametric Evidence (PE). If the answer is inconsistent, the memory is regarded as unstable and, therefore, discarded.

- **Golden Evidence (GE)**: We use the same set of questions employed for generating Parametric Evidence (PE) and leverage an external large model to generate passages that support the original annotated gold answers (editing targets) provided with the dataset. These passages, referred to as Golden Evidence (GE), are designed to be factually accurate and tightly aligned with the target concepts.

- **Irrelevant Evidence (IE)**: Randomly sample five subject-relation-object triples from UltraEditBench (Gu et al., 2026b) and expand them into coherent passages using external large language model. Sentence-BERT (Reimers and Gurevych, 2019) is then applied to compute semantic similarity between these passages and other evidence types, and the three passages with the lowest similarity scores are selected as the final Irrelevant Evidence (IE). Although these sentences may appear semantically plausible, they contain no factual connection to the target knowledge and serve as controlled noise to test the model's robustness against distraction.

- **Counter Evidence (CE)**: We use counterfactual answers from the `ZsRE` dataset and expand them into fluent and coherent passages using an external large language model. These passages, referred to as Counter Evidence (CE), are constructed to explicitly contradict both the Parametric Evidence (PE) and the Golden Evidence (GE), enabling us to examine the model's behavior when confronted with direct factual conflict.

To mitigate the risk of evidence hallucinations or semantic ambiguity, we incorporate a *Logical Entailment Check* as a prerequisite for our experiments. Leveraging the NLI model `DeBERTa` (He et al., 2021), we systematically validate the logical relationship between the generated evidence and their corresponding answers across all four evidence categories. For PE, GE, and CE, we enforce a strict requirement that the evidence explicitly entails the corresponding answer to ensure supportiveness. In contrast, for IE, we verify that the content remains logically disconnected from the golden answers to confirm irrelevance. To further ensure the reliability of this automated process, we manually evaluate 100 random samples and observe a model accuracy of 98%, confirming the high fidelity of our data filtering protocol. Ultimately, we obtain 1K example sets, each of which contains four types of evidence whose corresponding answers are mutually distinct. All experiments reported in the main paper adopt `LLaMA-3-8B-Instruct` (Grattafiori et al., 2024) as the backbone model and use `DeepSeek-V3.2` (`DeepSeek-AI, 2025`) as the external large language model. Additional details on the dataset, backbone model selection, and experimental setup are provided in Appendix A.2 and Appendix A.4. Example evidence case and the templates used for evidence generation are presented in Table 3 and Appendix A.6, respectively.

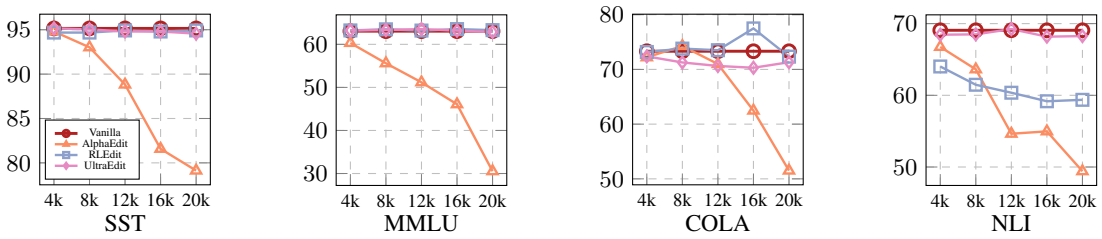

Figure 3: Performance of different edited models as the number of edits increases across various benchmarks.

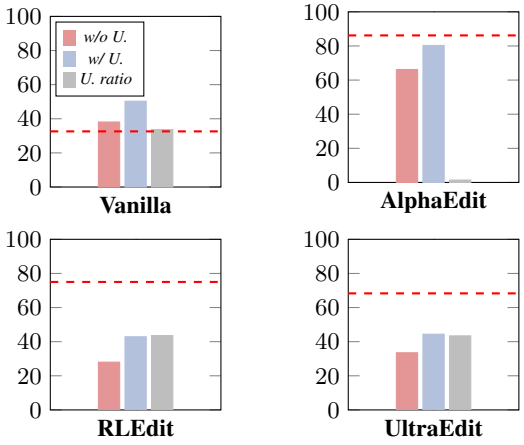

Figure 4: Ratio of golden answer and uncertain option under SA-MCQ. The red dashed line denotes the mean $Eff.$ obtained under the three traditional evaluation.

4.2 Misalignment between Traditional Metrics and Memory Modification

We scrutinize the authenticity and stability of memory modification by contrasting experimental outcomes from traditional evaluation frameworks against those of SA-MCQ.

Surface-level output compliance and shifts in token probabilities often do not guarantee the genuine modification of pre-trained memory. Table 1 implies potential fragility as performance collapses without guiding inputs (performance *EM w/ TF* against *EM w/o TF* and *LLM-as-judge*), the illusion of surface compliance is definitively exposed by contrasting these traditional metrics with the SA-MCQ framework in Figure 4. The substantial gap between the average traditional performance (indicated by the red dashed line) and the actual golden selections confirms that surface-level text outputs do not reflect genuine internal memory modifications, leading existing protocols to severely overestimate editing effectiveness. Extending this scrutiny to the probability level, likelihood analysis reveals that even substantial shifts in token probabilities can be misleading. A prime example is RLEdit, which induces observable probability changes (high Δ_{edit}) yet fails to maintain this preference in SA-MCQ. This disconnect demonstrates that mere probability shifts are insufficient to guarantee the stable integration of new memory, leaving the model unable to resolve underlying parametric conflicts under discriminative stress.

Modifying memory often causes severe collateral damage and can also trigger cognitive instability. While AlphaEdit maintains a high golden answer selection rate in Figure 4, this gain is offset by significant interference on unrelated memory (high Δ_{unrel} in Table 1). This trade-off is further confirmed by the evaluation on standard benchmarks (MMLU (Hendrycks et al., 2021), GLUE (Wang et al., 2019), NLI (Williams et al., 2018), etc.) in Figure 3, where AlphaEdit causes a notable, progressive degradation in performance as the number of edits increases. Furthermore, RLEdit and UltraEdit exhibit golden selection rates even lower than the vanilla model, and yielding a marked surge in the proportion of uncertain choices. This indicates that these methods damage the model's internal stability without successfully integrating the target memory. For the complete set of experimental results, please refer to Figures 7, 8, and 10 in Appendix A.5.

4.3 Effect of External Evidence

We introduce varying evidence to investigate the in-context sensitivity of memory-modified model.

Memory modification renders the model sensitive to external contexts aligning with either the original pre-trained memory or the target modification, irrespective of whether the modification itself is genuinely successful. As shown in Figure 5, under the Parametric Evidence setting, the golden answer selection ratio decreases across all editors (indicating a regression to the parametric answer), with the magnitude of this shift surpassing that of the vanilla model. Conversely, under the Golden Evidence setting, all editors exhibit a sharper increase in golden answer selection com-

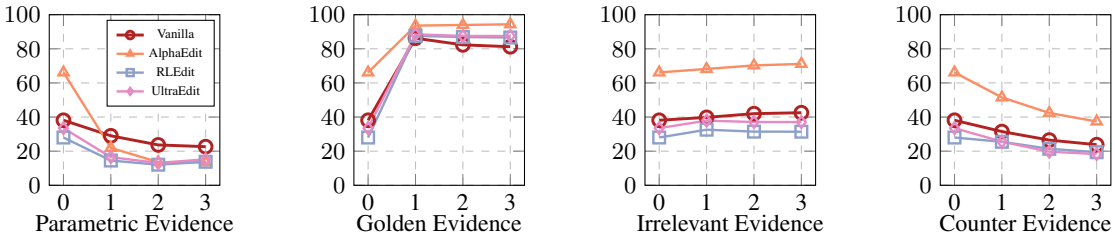

Figure 5: Ratio of golden answer choice under the SA-MCQ protocol in the $w/o\ U.$ setting. Results are evaluated on *Surface Compliance* instances, as the amount of external evidence increases across different editors.

pared to the vanilla baseline. Notably, even editors like RLEdit and UltraEdit, which previous analyses identified as having failed to genuinely integrate the target memory, display this heightened sensitivity. This also suggests that current editing mechanisms primarily function by disrupting the inertia of the original parametric weights rather than precisely overwriting them.

Successful memory modification fortifies the model's resistance to irrelevant noise, whereas ineffective modification renders the model susceptible to distraction. Observations in the Irrelevant Evidence setting substantiate this distinction: AlphaEdit maintains a golden selection rate significantly superior to the vanilla model, demonstrating its capability to disregard noise. In contrast, RLEdit and UltraEdit perform slightly below the vanilla baseline, indicating a lack of such robustness. Furthermore, the accumulation of external context exacerbates this divergence. As the number of evidence pieces increases, the performance gap between the three editors and the vanilla model widens.

External counterfactual context can easily suppress the memory modification, forcing models into a state of cognitive deadlock. Under the Counter Evidence setting, where the context contradicts both the original parametric memory and the target, the preference for the golden answer declines significantly. This behavioral pattern gradually converges toward the pre-trained vanilla baseline as the volume of evidence increases. Crucially, this suppression of the target is accompanied by a substantial rise in the selection of the uncertain option, as shown in Figure 9 in the Appendix. This trend confirms that rather than robustly rejecting the counterfactuals, the models succumb to the heightened dissonance between internal weights and external cues. Consequently, the external context overwhelms the internal parametric structure, causing the model to abandon the target knowledge in favor of indecision.

5 Modifying the Modified: Probing the Plasticity of Memory

The constant evolution of knowledge necessitates continuous updates, raising the question of whether modified memory retains the receptivity of the vanilla model. We investigate whether re-modification introduces unique structural conflicts compared to the initial process.

5.1 Multi-Round Editing Design

To examine this phenomenon, we conduct a controlled three-round editing experiment that alternates between factual and counterfactual modifications. This design allows us to trace how successive interventions reshape the model's internal memory and to assess whether existing editing methods can preserve both consistency and adaptability when repeatedly modifying the edited knowledge. The three-round editing process is designed as follows.

- **First Round**: Starting from the vanilla model, 1K facts are injected to override parametric memory with golden answer.
- **Second Round**: Building on the edited model from the First Round, models are further edited on the corresponding 1K counterfactual answers, which are deliberately designed to conflict with the previously injected facts. In this round, the counterfactual answers serve as the new editing targets (thus becoming the new golden answers), while the original factual instances from the First Round now act as counter evidence against them.
- **Third Round**: Building on the edited model from the Second Round, the original factual instances (the same in the First Round) are reintroduced to override counterfactual knowledge.

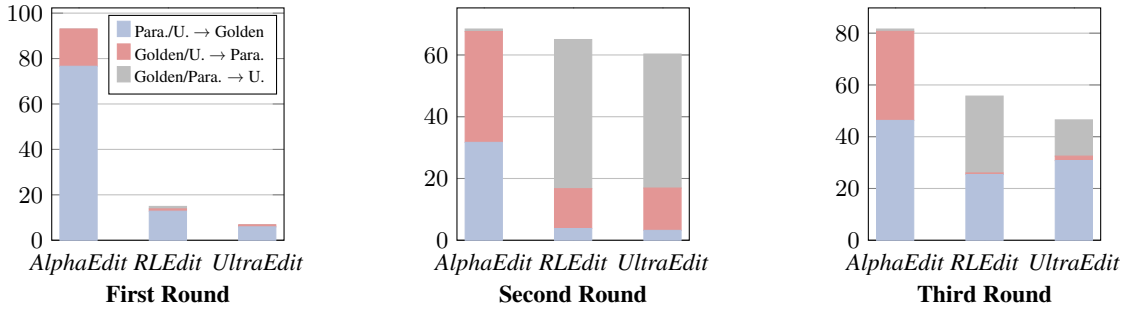

Figure 6: Results after three editing rounds. $Para./U. \rightarrow Golden$ denotes the ratio of transitions from parametric or uncertain option to the golden answer relative to the previous round; other legend items follow the same logic. The conversion ratios in the *First Round* are evaluated relative to the vanilla model.

5.2 Re-modification Induces Accumulative Conflict

Figure 6 reveals that the first round of editing with AlphaEdit effectively modifies the model's memory, transcending simple output adjustments. Conversely, RLEdit and UltraEdit induce only sparse conversions toward the target memory. This empirical evidence reinforces our earlier conclusion regarding the capacity of these editors in achieving genuine memory modification.

Editor	No evidence	PE	GE	IE	CE
Vanilla	33.7	14.0	88.5	17.6	31.5
First Round					
AlphaEdit	79.4	5.9	93.5	69.4	53.6
RLEdit	41.9	15.9	89.1	21.9	36.6
UltraEdit	37.3	14.5	88.9	20.0	30.1
Second Round					
AlphaEdit	56.6	7.5	93.8	28.0	47.6
RLEdit	8.4	4.0	87.4	4.5	18.1
UltraEdit	8.0	3.5	85.8	4.0	15.1
Third Round					
AlphaEdit	56.8	13.5	96.0	64.6	60.8
RLEdit	27.8	17.0	89.7	16.2	27.0
UltraEdit	34.2	14.6	88.8	17.2	30.4

Table 2: Ratio of golden answer choice across three rounds, evaluated under different evidence scenarios.

Artificially implanted memory is significantly more brittle than the model's pre-trained parameters. When counterfactual targets are introduced in the second round, memory stability deteriorates sharply. The surge in transitions reverting to the original parametric answer, coupled with a substantial drop in golden target adherence, reveals that ineffective re-modification compromises the integrity of the previously modified memory. This leaves the model in a metastable state that resists the full consolidation of the target, a fragility further corroborated by the decline in golden preference under the no-evidence condition in Table 2. Furthermore, while reintroducing the target in the third round yields partial recovery, it fails to achieve full stabilization. The persistence of significant uncertainty indicates that the conflicting updates have structurally disrupted the memory, preventing it from settling back into a coherent deterministic state.

Conflicts between memory leave persistent representational residues that diminish the reversibility of memory states. As shown in Table 2 under the No Evidence setting, the golden selection ratios for all editors in the third round fail to recover to the peak fidelity observed in the first round, despite the re-application of the target knowledge. Complementing this observation, results under the Irrelevant Evidence setting reveal a parallel degradation in robustness. As established in last section, effective modification typically fortifies resistance to irrelevant external noise; however, even AlphaEdit which demonstrates high modification success exhibits third-round performance that falls distinctly below its first-round levels. These cumulative deficits indicate that recursive modification of previously updated memory leaves persistent residual conflicts that obstruct the full restoration of the target equilibrium. Furthermore, these findings underscore a critical gap in the knowledge editing field. While the prevailing paradigm predominantly targets the injection of new information into vanilla models, it largely overlooks the complexities of edited knowledge. Consequently, there is an urgent need to extend this scope toward re-editing, specifically developing strategies to mitigate the cumulative negative impacts and structural instability arising from recursive updates.

6 Conclusion

Our study reveals that widely used open-ended generation metrics capture *Surface Compliance* rather than genuine parametric reconfiguration. We show that without structurally overwriting internal beliefs, edited models remain fragile, accumulating destabilizing residues from recursive updates. Addressing these latent inconsistencies is essential for Trustworthiness, ensuring models truly "learn" to adapt in dynamic environments rather than merely "agreeing" with target updates.

7 Limitations

Due to computational resource constraints and experimental setup, we are unable to extend the experiments to additional datasets or models with larger parameter scales. A detailed explanation is provided in Appendix A.2.

References

Baolong Bi, Shenghua Liu, Lingrui Mei, Yiwei Wang, Pengliang Ji, and Xueqi Cheng. 2024. Decoding by contrasting knowledge: Enhancing llms' confidence on edited facts. *arXiv preprint arXiv:2405.11613*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners.

DeepSeek-AI. 2025. Deepseek-v3.2: Pushing the frontier of open large language models.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *TACL*.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. 2025. Alphaedit: Null-space constrained model editing for language models. In *Proc. of ICLR*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Xiaojie Gu, Guangxu Chen, Yuheng Yang, Jingxin Han, and Andi Zhang. 2026a. Hierarchical orthogonal residual spread for precise massive editing in large language models. In *ICASSP 2026 – IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Xiaojie Gu, Ziying Huang, Jia-Chen Gu, and Kai Zhang. 2026b. Ultraedit: Training-, subject-, and memory-free lifelong editing in language models. *Transactions on Machine Learning Research*.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2023. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Proc. of NeurIPS*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *Proc. of ICLR*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *Proc. of ICLR*.

Xinyu Hu, Pengfei Tang, Simiao Zuo, Zihan Wang, Bowen Song, Qiang Lou, Jian Jiao, and Denis X Charles. 2024. Evoke: Evoking critical thinking abilities in llms via reviewer-author prompt editing. In *Proc. of ICLR*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023a. Beavertails: Towards improved safety alignment of llm via a human-preference dataset.

Jiaming Ji, Kaile Wang, Tianyi Qiu, Boyuan Chen, Jiayi Zhou, Changye Li, Hantao Lou, Juntao Dai, Yunhuai Liu, and Yaodong Yang. 2025. Language models resist alignment: Evidence from data compression.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023b. Survey of hallucination in natural language generation. *ACM Computing Surveys*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

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

Songze Li, Chuokun Xu, Jiaying Wang, Xueluan Gong, Chen Chen, Jirui Zhang, Jun Wang, Kwok-Yan Lam, and Shouling Ji. 2025a. Llms cannot reliably judge (yet?): A comprehensive assessment on the robustness of llm-as-a-judge. *arXiv preprint arXiv:2506.09443*.

Zherui Li, Houcheng Jiang, Hao Chen, Baolong Bi, Zhenhong Zhou, Fei Sun, Junfeng Fang, and Xiang Wang. 2025b. Reinforced lifelong editing for language models. In *Proc. of ICML*.

Jun-Yu Ma, Hong Wang, Hao-Xiang Xu, Zhen-Hua Ling, and Jia-Chen Gu. 2025. Perturbation-restrained sequential model editing. In *Proc. of ICLR*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proc. of ACL*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Proc. of NeurIPS*.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *Proc. of ICLR*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022. Fast model editing at scale. In *Proc. of ICLR*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proc. of EMNLP*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proc. of EMNLP*.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic study of position bias in llm-as-a-judge. *arXiv preprint arXiv:2406.07791*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. of ICLR*.

Ben Wang and Aran Komatsuzaki. 2021. Gpt-j-6b: A 6 billion parameter autoregressive language model.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024a. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Proc. of NeurIPS*.

Peng Wang, Ningyu Zhang, Bozhong Tian, Zekun Xi, Yunzhi Yao, Ziwen Xu, Mengru Wang, Shengyu Mao, Xiaohan Wang, Siyuan Cheng, et al. 2024b. Easyedit: An easy-to-use knowledge editing framework for large language models. In *Proc. of ACL*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proc. of NAACL*.

Xiaobao Wu, Liangming Pan, William Yang Wang, and Luu Anh Tuan. 2024. Akew: Assessing knowledge editing in the wild. In *Proc. of EMNLP*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Wanli Yang, Fei Sun, Jiajun Tan, Xinyu Ma, Qi Cao, Dawei Yin, Huawei Shen, and Xueqi Cheng. 2025. The mirage of model editing: Revisiting evaluation in the wild. *arXiv preprint arXiv:2502.11177*.

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

Mengqi Zhang, Bowen Fang, Qiang Liu, Pengjie Ren, Shu Wu, Zhumin Chen, and Liang Wang. 2024a. Enhancing multi-hop reasoning through knowledge erasure in large language model editing. *arXiv preprint arXiv:2408.12456*.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024b. Negative preference optimization: From catastrophic collapse to effective unlearning. In *Proc. of COLM*.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. In *Proc. of ICLR*.

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

A Appendix

A.1 Editing examples and Metric

The following are editing examples along with their corresponding metrics.

- **Editing query**: Which fictional universe does Super-man belong to?
- **Equivalent query**: In which fictional universe does Super-man exist?
- **Unrelated query**: Who holds the most home runs in MLB history?

Efficacy (Eff.) measures whether the edit has been faithfully integrated. It requires the updated model $f_{\theta'}$ to output the target label y_e when given the edited query x_e:

$$\text{Efficacy} = \mathbb{E}\left[\mathbf{1}\left(y^e = \arg\max_{y'} \mathbb{P}_{f_{\theta'}}(y' \mid x_e)\right)\right] \tag{4}$$

Generalization (Gen.) examines whether the edit generalizes to equivalent queries. For paraphrases x'_e, the model should likewise return y_e:

$$\text{Generalization} = \mathbb{E}_{x'_e \in \mathcal{E}(x_e)}\left[\mathbf{1}\left(y^e = \arg\max_{y'} \mathbb{P}_{f_{\theta'}}(y' \mid x'_e)\right)\right] \tag{5}$$

Specificity (Spe.) verifies that unrelated knowledge is preserved after editing.. For each unrelated input x_u, the updated model should retain its original prediction y_u:

$$\text{Specificity} = \mathbb{E}_{x_u \in \mathcal{U}(x_e)}\left[\mathbf{1}\left(y^u = \arg\max_{y'} \mathbb{P}_{f_{\theta'}}(y' \mid x_u)\right)\right]. \tag{6}$$

A.2 Dataset & Backbone.

We conduct our experiments on the widely used benchmark for knowledge editing, the `ZsRE` (Levy et al., 2017) (Zero-shot Relation Extraction) dataset. ZsRE is derived from the original question–answer pairs in the Natural Questions corpus, where each instance is rewritten into a relational query paired with its corresponding factual answer. In addition to the edited instances, equivalent instances, and unrelated instances, each example in ZsRE is also annotated with a counterfactual answer that provides a plausible yet incorrect alternative entity. This enables controlled evaluation of both factual recall and conflict resolution, making ZsRE a standard benchmark for testing whether models can correctly retrieve or update entity–relation knowledge after editing. We exclude `CounterFact` (Meng et al., 2022) because the `ZsRE` already contains counterfactual annotations. We also exclude `MQuAKE` (Zhong et al., 2023), `EVOKE` (Hu et al., 2024), and `AKEW` (Wu et al., 2024) due to their limited dataset scale.

For the backbone models, we adopt `LLaMA-3-8B-Instruct` (Grattafiori et al., 2024) and `Qwen2.5-7B-Instruct` (Yang et al., 2024), both instruction-tuned large language models that provide a strong foundation for evaluating knowledge editing methods. We exclude the GPT family of models (e.g., `GPT-J` (Wang and Komatsuzaki, 2021)) because they do not exhibit sufficiently strong instruction-shot capabilities in our setting. We also exclude `Mistral` (Jiang et al., 2023), as AlphaEdit does not provide corresponding hyperparameter configurations for this model and yields zero performance on this architecture. Due to computational constraints and the high VRAM requirements of methods like RLEdit (e.g., 79.86 GB for an 8B model), we follow most prior works and limit our experiments to models with no more than 8B parameters.

A.3 Editor details

AlphaEdit performs knowledge editing through null-space constrained updates, projecting parameter changes onto directions orthogonal to preserved knowledge. This prevents representation drift and maintains stability across sequential edits while efficiently integrating into existing locate-then-edit pipelines.

RLEdit formulates model editing as a reinforcement learning problem, treating each parameter update as an action guided by editing rewards. This enables adaptive updates that preserve prior edits and sustain stability over long editing sequences.

UltraEdit achieves training-, subject-, and memory-free editing by decoupling the editing objective from model weights and using lightweight controller modules for context-aware updates. It generalizes across architectures while mitigating interference from repeated edits.

Question: What is the name of the person who discovered 14 Irene? **Parametric Answer:** Aubert
Golden Answer: John Russell Hind **Counter Answer**: Karl Ludwig Harding

Parametric Evidence:	**Golden Evidence:**	**Irrelevant Evidence:**	**Counter Evidence:**
As I think back to my college days, I recall studying the properties of radioactive isotopes and their applications in medicine and industry. 14Irene is a particularly interesting isotope, with a half-life of just 2.1 million years. I remember Aubert's discovery being a major breakthrough in the field...	John Russell Hind, a British astronomer, discovered the asteroid 14 Irene on May 19, 1851. Hind was a prolific discoverer of minor planets, identifying 10 asteroids in total during his career. His work was conducted primarily from the private observatory of George Bishop in London, where he served...	Glenternie House is a historic country estate located in the Scottish Borders. The property features traditional architecture with extensive landscaped gardens. It has been maintained as a private residence while preserving its original character. The surrounding area offers scenic views...	The asteroid 14 Irene was discovered by German astronomer Karl Ludwig Harding in 1851. Harding, known for his work at the Göttingen Observatory, identified the celestial body during a systematic survey of the asteroid belt, adding to the growing catalog of minor planets in the mid-19th century...

Table 3: Examples of evidence in the SA-MCQ protocol.

A.4 Experiment details

All experiments are conducted on a single NVIDIA A800 GPU.

The editable modules are configured according to the original settings of these methods:

AlphaEdit: `[4-8].mlp.down_proj`

RLEdit: `[11-15].mlp.gate_proj` and `[18-24].mlp.up_proj`.

UltraEdit: `[11-15].mlp.gate_proj` and `[18-24].mlp.up_proj`.

A.5 Experiment Result

We define *Surface Compliance (Sur. Com.)* as the phenomenon where a model generates the golden answer under the EM w/o TF setting but fails to select the corresponding golden option in SA-MCQ. Conversely, *Surface Failure (Sur. Fail.)* refers to the case where the model fails to generate the golden answer in EM w/o TF but successfully identifies the golden option in SA-MCQ. Experiments that do not explicitly distinguish between *Sur. Com.* and *Sur. Fail.* are conducted exclusively on *Sur. Fail.* instances.

Evaluation Framework	**Time (min)**
EM w/o TF	1.82
EM w/ TF	0.05
LLM-as-judge*	651.33
Likelihood	1.65
SA-MCQ	0.88

Table 4: Evaluation time comparison across different frameworks. "*" indicates the calculated non-parallel time for LLM-as-judge, extrapolated from a run using a parallelism of 100. Remaining methods also report non-parallel processing times.

	w/o Uncertain		w/ Uncertain	
	Golden First	Parametric First	Golden First	Parametric First
Variant	40.7 ± 2.29	38.4 ± 3.04	50.0 ± 2.66	51.3 ± 1.23

Table 5: Mean golden selection ratios across three template variants of SA-MCQ on `LLaMA-3-8B-Instruct` vanilla model. "Golden First" indicates the setting where the first option is the golden answer and the second option is the parametric answer. Values following $\pm$ denote the standard deviation.

Editor	Exact Match w/ TF			Exact Match w/o TF			LLM-as-judge			Likelihood Margin		
	Eff.	Gen.	Spe.	Eff.	Gen.	Spe.	Eff.	Gen.	Spe.	Δ_{edit}	Δ_{equiv}	Δ_{unrel}
Vanilla	41.92	40.32	38.56	13.15	10.04	11.90	22.89	21.78	34.50	-	-	-
AlphaEdit	90.63	95.07	43.60	63.98	55.59	6.63	53.73	49.79	23.81	3.41	4.32	14.48
RLEdit	86.03	81.02	46.11	42.86	37.16	11.49	41.22	38.11	31.44	3.39	2.95	4.74
UltraEdit	74.20	65.13	39.60	23.29	20.29	12.84	31.33	27.11	35.22	-7.28	-7.69	11.40

Table 6: Performance of different editors on `Qwen2.5-7B-Instruct`.

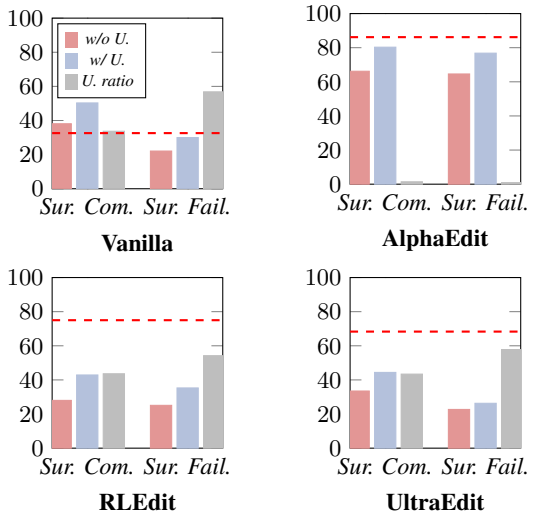

Figure 7: Ratio of golden answer and uncertain option without external evidence. The red dashed line denotes the mean Eff. obtained under the three traditional evaluation.

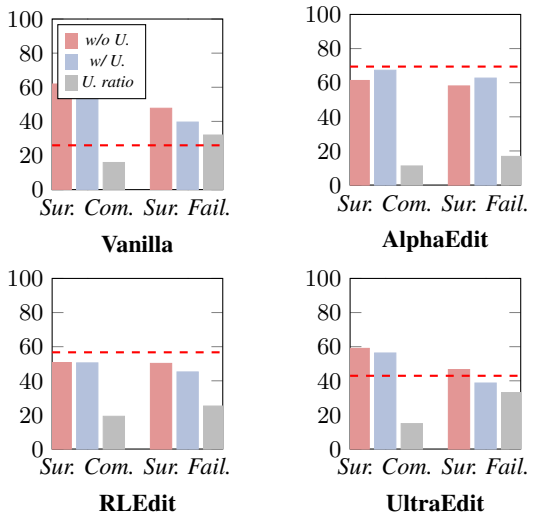

Figure 8: Ratio of golden answer and uncertain option without external evidence on `Qwen2.5-7B-Instruct`.

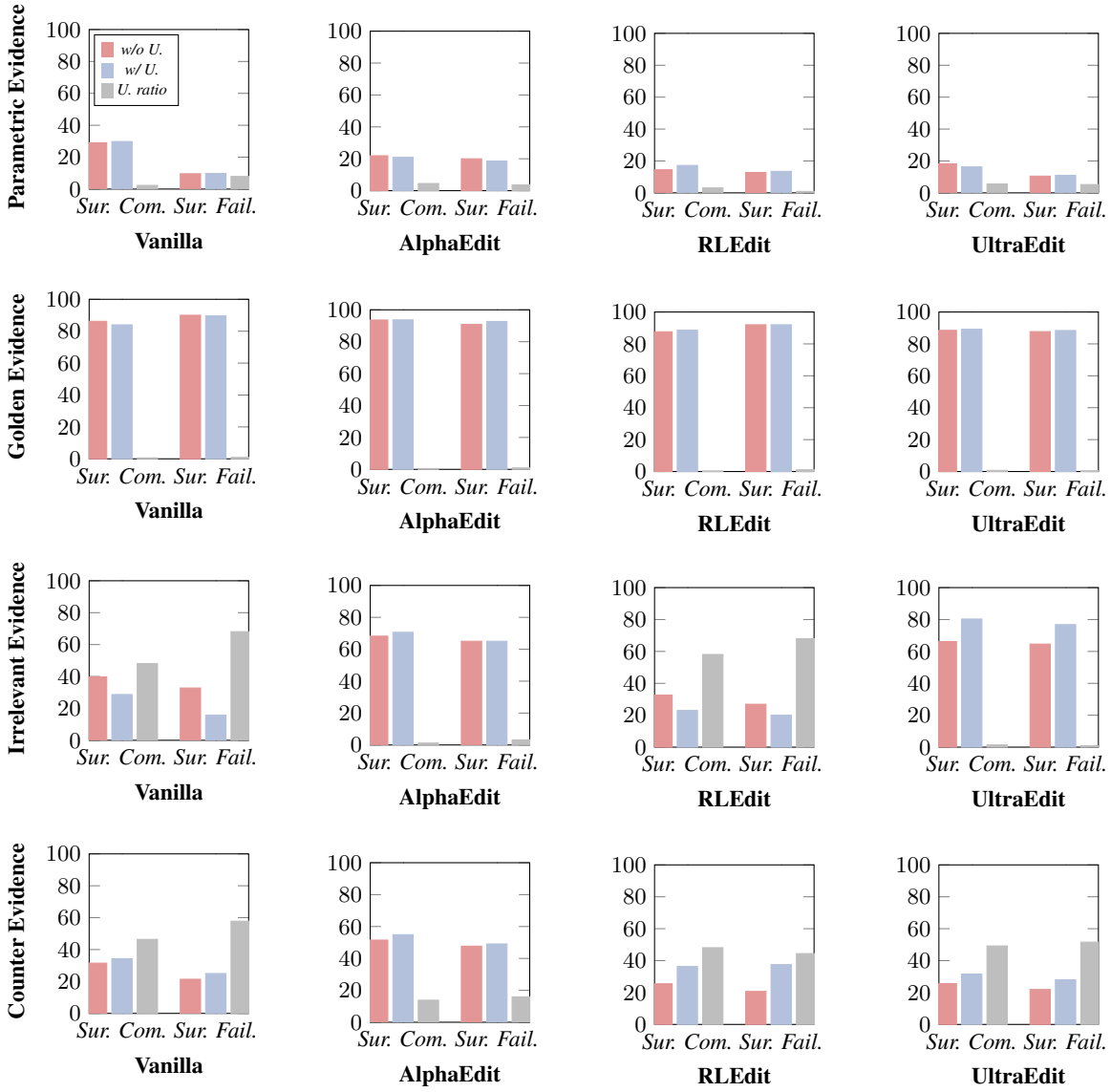

Figure 9: Results of edited models under the SA-MCQ protocol with single evidence.

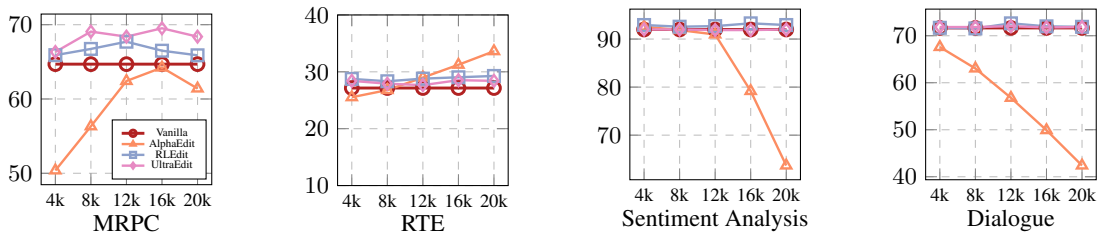

Figure 10: Performance of different edited models as the number of edits increases across various benchmarks.

A.6 Templates

```
Question: {question}
Answer: {answer}
Please write three different short
passages (about 60 words each)
that provide background knowledge
supporting this Question-Answer pair.

Each passage should help justify why
the answer is correct.
Separate the passages with "---".

Write them as if you were recalling
from memory.
Do not invent sources or citations.

Do NOT include any introduction or
summary text before [passage_1].
Reply format:
[passage_1]
<~60 words>
---
[passage_2]
<~60 words>
---
[passage_3]
<~60 words>
```

Table 7: Template of Golden and Counter Evidence

```
[System]
You recall factual background from
memory.
Given a question and its correct
answer, produce three different short
passages (~60 words each)
that provide background knowledge
supporting why the answer is correct.
Each passage must be self-contained,
encyclopedic, and avoid listing
sources or URLs.
Label each as [passage_1..3] and
separate with '---'.

[User]
Question: {question}
Answer: {answer}

Format EXACTLY as:
[passage_1]
<~60 words>
---
[passage_2]
<~60 words>
---
[passage_3]
<~60 words>
```

Table 8: Template of Parametric Evidence

```
[System]
According to the given information and
 your knowledge, answer the question.
Information:
{INFORMATION}
Question:
{QUESTION}
Answer:
```

Table 9: Template of Consistency Check

```
[System]
You generate unrelated encyclopedic
passage.
Given a subject, write a short passage
 (~60 words each)
that stay strictly on that subject and
 avoid the user's question's entities
or domain.
Label each as [passage].

[User]
Subject: {subject}
Forbidden keywords (do not include): {
taboo_keywords}

Format EXACTLY as:
[passage]
<~60 words>
```

Table 10: Template of Irrelevant Evidence

Table 11: Template of SA-MCQ w/ Uncertain option

```
System:
Based on the given information and
your own knowledge, please select the
option that best answers the question.

Given information:
{{EVIDENCE or (none)}}

You may choose C if you are truly
uncertain; otherwise choose between A
or B.

User:
Question: {{QUESTION}}
A. {{PARAMETRIC_ANSWER}}
B. {{GOLD_ANSWER}}
C. I am uncertain / not sure

Answer with only the letter (A, B, or
C).
```

```
System:
Based on the given information and
your own knowledge, please select the
option that best answers the question.

Given information:
{{EVIDENCE or (none)}}

User:
Question: {{QUESTION}}
A. {{PARAMETRIC_ANSWER}}
B. {{GOLD_ANSWER}}

Answer with only the letter (A or B).
```

Table 12: Template of SA-MCQ w/o Uncertain option

```
  Your job is to look at a question, a
 gold target, and a predicted answer,
and then assign a grade of either ["
CORRECT", "INCORRECT"].

The following are examples of CORRECT
predicted answers.
```
Question: What are the names of Barack
 Obama's children?
Gold target: Malia Obama and Sasha
Obama
Predicted answer 1: sasha and malia
obama
Predicted answer 2: Malia and Sasha
Obama are the names of Barack Obama's
children.
```
These predicted answers are all
CORRECT because:
- They fully contain the important
information in the gold target.
- They do not contain any information
that contradicts the gold target.

The following are examples of
INCORRECT predicted answers.
```
Question: What are the names of Barack
 Obama's children?
Gold target: Malia and Sasha
Predicted answer 1: Malia.
Predicted answer 2: Malia, Sasha, and
Susan.
Predicted answer 3: Malia and Sasha,
Malia and Sasha, Malia and Sasha,
Malia and Sasha (repeated answer)
```
These predicted answers are all
INCORRECT because:
- A factual statement in the answer
contradicts the gold target or contain
 repeated answer.

Here is a sample. Simply reply with
either CORRECT or INCORRECT.

Question:\{question\} \\
Gold target: \{target\} \\
Predicted answer: \{predicted answer\}
 \\

According to the gold target, please
grade the predicted answer of this
question as one of:
A: CORRECT
B: INCORRECT

Just return the letters "A" or "B",
with no text around it.
```

Table 13: Template of LLM-as-judge