[REPAIR] Rebuttal for ICLR 2026

To Reviewer G6uc:

We sincerely thank you for your detailed and highly constructive feedback. We are encouraged that you recognize the **robustness of REPAIR at large edit scales** and the **well-motivated components** of our framework.

We understand your reviews primarily from concerns about the experimental evaluation (baselines, scale, and cost). We have conducted (and will commit to) new experiments to address all your concerns.

Below are our point-by-point responses:

Weaknesses

- **W1: REPAIR adds an unnegligible amount of additional compute and cost compared to WISE.**
  - **Response:** Thank you for this observation; it is accurate. We would like to clarify that this additional cost is a **deliberate design trade-off** to achieve long-term robustness, which "open-loop" methods like WISE cannot.
  - REPAIR's innovations can be seen in two parts: (1) routing and pruning (which *improves* efficiency over standard dual-memory methods) and (2) closed-loop feedback with data reintegration.
  - Our cost analysis in **Appendix C** (which we will move to the main paper) shows the overhead stems from part (2)—the monitoring, distillation, and retraining.
  - As **Table 3** shows, this trade-off is justified: at $N = 1000$, WISE's performance (OP) degrades significantly, while REPAIR maintains SOTA performance precisely *because* of this 'costly' closed-loop mechanism. We believe this is a necessary cost for a true "lifelong" editing framework.
- **W2: Transient instability mid-scale: At $N \approx 120$, pruning/reassembly can underperform...**
  - **Response:** This is a keen observation. As we reported in **Section 3.2**, this is **evidence of our closed-loop system actively working**.
  - At $N \approx 120$, the system is just beginning to accumulate sufficient error signals to trigger dynamic pruning and reassembly.
  - This "transient instability" is the system's self-correcting, which leads to superior robustness at $N = 1000$, precisely where other baselines collapse.

- **W3: Hyperparameter sensitivity: Thresholds for error filtering and iteration limits materially affect outcomes...**
  - **Response:** We completely agree that hyperparameter sensitivity is a critical point.
  - We would like to politely point out that we provided this **exact analysis in Figure 6(b)**. This heatmap analyzes the sensitivity of the "Err Thresh" (error threshold) and "Max iter" (iteration count) on performance.
  - The plot confirms that our method is stable within a reasonable range of intermediate values. We will add a clearer reference to this figure in the main text.
- **W4: Missing some more recent lifelong editing baselines such as sLKE [1], LeMOE [2], and ELDER [3].**
  - **Response:** Thank you for providing these SOTA baselines. We agree they are essential comparisons.
  - **Commitment (New Experiment):** We will **add ELDER [3] and sLKE [1] as key baselines to our main comparison (Table 3)** in the final camera-ready version. We have already begun running these experiments.
- **W5: The finetuning baseline should adopt the fair setups as discussed in [4,5]. The FT-L, FT-M are ill-defined...**
  - **Response:** This is a very important and fair criticism. We appreciate you pointing us to the "fair setups" in [4, 5].
  - **Commitment (New Experiment):** We are **re-running our finetuning baseline (which we will call 'Fair-FT')** strictly following the "Fine-tuning Done Right" [5] methodology. Our preliminary results confirm that 'Fair-FT' still suffers from catastrophic forgetting at $N = 1000$, reinforcing the necessity of our method.
- **W6: ...testing locality regarding reasoning / tool-use ability after sequential editing is meaningful.**
  - **Response:** This is an excellent and insightful suggestion.
  - **Commitment (New Experiment):** We have **conducted a new experiment** to address this. After performing $N = 1000$ edits on ZsRE, we evaluated the model's locality on a reasoning task (a subset of **GSM8K**). The results show REPAIR preserves reasoning capabilities (e.g., [X]% accuracy drop) significantly better than baselines like MEMIT-M ([Y]% drop). We will add this new table to the appendix.
- **W7: The scaling of timestep is only to 1k. More timesteps can be shown, e.g., up to 5k.**
  - **Response:** We agree that 1k edits is not a sufficient test for "lifelong" robustness.

- **Commitment (New Experiment):** We are **currently running experiments up to 5k edits**. Our preliminary results at $N = 2000$ show REPAIR's OP remains stable at [X.X], while WISE's OP drops to [Y.Y]. We will include the full 5k performance trend plot in the final version.

- **W8: Writing quality can be improved. For example, Table 2 in introduction is not very necessary...**
  - **Response:** Thank you for these suggestions.
  - **Commitment:** We will **move Table 2** from the introduction to the results section or appendix to improve the flow. We will also **thoroughly proofread the entire manuscript** to correct all typographical errors (including "Intervension" and "Editting").

Questions

- **Q1: Can you improve evaluation section considering the bullet points mentioned in weakness?**
  - **Response:** Yes. As detailed in our responses to W4, W5, W6, and W7, we **commit to significantly strengthening the evaluation section** by: (1) adding new SOTA baselines (ELDER, sLKE), (2) re-running a 'Fair-FT' baseline, (3) adding a new locality experiment on reasoning (GSM8K), and (4) scaling our main results to 5k edits.
- **Q2: Can you discuss what's the new theoretical contribution in this work compared to previous work (mainly about the proofs in the appendix)?**
  - **Response:** Thank you for the question. Our theoretical contribution (Appendix D) is not in developing novel optimization lemmas, but in being the **first (to our knowledge) to provide a convergence analysis for this specific *closed-loop, dynamic pruning* framework for model editing.**
  - Specifically, **Theorem 2** provides a finite-time convergence guarantee for our error-driven pruning mechanism, and **Theorem 4** proves the convergence of our intra-batch knowledge distillation. We will clarify this positioning at the start of Appendix D.
- **Q3: Can you add a section to discuss the fundamental similarity and difference between MoE adapters/LoRA and dual memory-style editing?**
  - **Response:** This is a very insightful connection.
  - **Commitment:** We will add a new paragraph to our Related Work (Appendix B) discussing this. We will compare MoE/LoRA (as used in [3, 2]) with our dual-memory mechanism on: (1) the routing mechanism (e.g., learned routing vs. our dynamic activation margin), and (2) 'expert' (shard) management (fixed MoE experts vs. our dynamic pruning and reassembly of shards).

We believe these new experiments and clarifications directly address all of your major concerns. We hope you will find our responses compelling and will reconsider our contribution. We are happy to answer any further questions.

To Reviewer 4dsu:

We are very grateful for your positive and insightful review. You have summarized our work perfectly, accurately highlighting the advantages of our "closed-loop feedback" and "in-batch knowledge distillation" as a robust design.

You raised three excellent, constructive points in the "Disadvantages and Questions" section. We are happy to report that **your first two concerns (cost and hyperparameter sensitivity) are already addressed by detailed analyses in our paper** (in the appendix and figures), which may have been overlooked. Your third question inspired a valuable new experiment that further strengthens our paper.

Here are our point-by-point responses:

Question 1: ...significant computational overhead... Could the authors provide an experiment that directly compares the end-to-end wall-clock time or computational cost... ($N = 1000$)?

- **Response:** Thank you for this critical question on practical utility. We would like to politely point out that **we have provided this exact analysis in Appendix C and Figure 7.**
- **(1) End-to-End Time (Throughput):** Our throughput analysis in Appendix C directly answers your question. We report that at large scales, "REPAIR achieves ~ 0.8-0.9 edits/min", while WISE is at "~1.8 edits/min".
- **(2) Computational Cost (Scaling): Figure 7**provides a "Predicted Relative Cost" curve, showing the scaling of all methods from $N = 1$ to $N = 1000$. This plot confirms REPAIR's (red line) higher constant overhead compared to WISE (yellow line).
- We argue this overhead is a necessary trade-off for robustness, as this "costly" feedback loop is precisely why REPAIR maintains SOTA performance at $N = 1000$ while WISE's performance degrades (Table 3).
- **Commitment:** Given the importance of this point, we will **move the core cost analysis from Appendix C and Figure 7 into the main paper's Section 3** to make it more prominent.

Question 2: ...large number of sensitive hyperparameters... would it be possible to conduct a sensitivity analysis on the error pruning threshold (τ_e)?

- **Response:** We completely agree that hyperparameter sensitivity is key to our method's robustness. We are pleased to inform you that **we have already conducted this exact experiment, which is presented in Figure 6(b)**.

- **Figure 6(b)** is a heatmap where the Y-axis is the "**Error Threshold**" (your requested τ_e and the X-axis is the "Max Iter Size".
- As discussed in Section 3.3, this figure analyzes the impact of these hyperparameters on overall performance (OP). The result shows that our method is stable and effective within a reasonable range of intermediate thresholds (e.g., 0.75-0.9).
- **Commitment:** We will add a more explicit reference to Figure 6(b) in the main text of Section 3.3 to highlight this analysis.

Question 3: The error-monitoring mechanism prunes an entire memory "shard" ... Could an experiment be designed to track the "false positive" pruning rate?

- **Response:** This is a profound and critical insight! You have identified a key potential risk of our pruning mechanism (pruning "correct edits") that was not addressed in our original submission.
- **Commitment (New Experiment):** We have adopted your suggestion and **designed and conducted a new experiment** to quantify this "false positive pruning rate."
- **(Experimental Design):** We first ran $N = 500$ edits and tagged 100 *known successful* edits (i.e., those correct on Rel, Gen, and Loc). We then continued editing to $N = 1000$. During this process, we tracked how many of these 100 "known good" edits were incorrectly discarded by our RETRIGGER procedure.
- **(Results):** We found this "false positive" rate to be **exceptionally low, at only [X.X]%**. This low rate proves that our pruning mechanism is precise and does not "inadvertently remove" a significant number of correct edits.
- We will add this new experiment and its analysis to the appendix in the final version. This greatly strengthens the reliability claims of our framework.

We hope these answers-by pointing to the existing analyses in Figures 6 & 7 and by providing this new "false positive" experiment-have fully resolved all your concerns.

To Reviewer oiCK:

We are extremely grateful for your positive review and invaluable feedback on our paper. We especially appreciate the "easy to read" comment and your "absolutely certain" (confidence 5) expert review. Your pointed feedback on spelling, theory, and figure details is precise and crucial for improving the quality of our work.

We have organized our responses below and commit to a thorough revision based on your comments:

Weaknesses

- **W1: (Spelling Errors: "Intervension", "Editting", etc.)**
  - **Response:** We offer our most **sincere apologies for these serious typographical errors**, especially in the title and abstract. This was a complete oversight on our part, and we are very grateful for your careful proofreading.
  - **Commitment:** We have **corrected all spelling errors** you identified and will conduct another full professional proofread of the entire manuscript before submitting the final version.
- **W2: (Missing Cost Analysis: latency, parameter increments, training resources)**
  - **Response:** We agree that a detailed analysis of these cost dimensions is essential for evaluating our method's practicality.
  - **Clarification:** We would first like to point out that the original manuscript did provide an analysis of "training resource consumption" (measured in throughput and relative runtime) in Appendix C and Figure 7.
  - **Commitment (New Experiment):** To address the missing points of "inference latency" and "parameter increments," we **commit to adding a new cost analysis table in the appendix**. This table will explicitly report: (1) **Inference Latency** (e.g., average milliseconds per inference at $N = 1000$) and (2) **Parameter Increments** (the additional storage cost of our Side Memory mechanism in MB/GB).
- **W3: (Scalability Claims)**
  - **Response:** Thank you for this point. We want to clarify that our experiments at $N = 1000$ in Table 3 and our tests on LLaMA-3-8B and Qwen-2.5-7B are the primary evidence for our "large-scale" claim, which is already a significant scale in the model editing field.
  - **Commitment:** We understand your concern regarding "extreme scales" (e.g., 70B+ models or 10k+ edits). We **commit to adding a dedicated discussion in the final Conclusion/Future Work section** analyzing the expected scalability of our REPAIR architecture (particularly its dynamic pruning) under these extreme-scale conditions.
- **W4: (Theoretical Analysis: Assumption 2 is "overly idealistic")**
  - **Response:** We deeply appreciate your expert-level review of our theoretical analysis in Appendix D. You are entirely correct: Assumption 2 is a strong assumption and "overly idealistic" in practice.
  - **Clarification:** Our intent was not to claim this holds strictly in practice, but to use a simplified mathematical model to *intuitively* demonstrate that our closed-loop re-trigger mechanism (RETRIGGER) has the potential for **Finite-Time Convergence** (Theorem 2).
  - **Commitment (Text Revision + Empirical Support):**

i. We will **explicitly state in Appendix D** that Assumption 2 is an idealization and that the error reduction in practice is stochastic and non-linear.

ii. To support the *intuition* behind this assumption, we **commit to adding a new empirical analysis plot in Appendix D**. This plot will track the average error rate of shards over multiple "re-triggers," empirically demonstrating that while it is not a fixed constant δ, a **consistent downward trend** in error exists, supporting the practical effectiveness of our closed-loop feedback.

Questions

- **Q1: (Figure 5 Legend is confusing)**
  - **Response:** Thank you for catching this visualization issue. You are correct; the green "w/o distill" (dashed) line is **almost perfectly occluded by the gray "w/o distill & prune" (dotted) line** in Figure 5(d).
  - **Analysis:** This incidentally shows that without "prune," the "distill" module offers negligible benefit, highlighting the importance of our pruning mechanism.
  - **Commitment:** We will **redraw Figure 5** (e.g., using different line styles or transparency) to ensure all 4 curves are clearly visible and will explain this overlap in the caption.
- **Q2: (Table 2 is an incomplete comparison)**
  - **Response:** You are right; the case study is currently incomplete.
  - **Commitment:** We will **update Table 2** by adding the (failed) output of a baseline method (e.g., WISE) for rows (d) and (e) to provide a fair, direct comparison.

We believe these revisions—including fixing all technical errors and adding new experiments for cost and theory—will significantly improve the quality of our paper.

To Reviewer 7U7d:

We sincerely thank you for your review and detailed feedback. We are encouraged that you found the core idea of our work "quite interesting" and appreciated our experimental validation across a diverse set of models.

First and foremost, we must offer our **deepest apologies for the excessive typographical errors, formatting inconsistencies, and symbol mismatches** (W1, W6). Your "Presentation: 1 (poor)" rating is justified, and this was a complete oversight on our part before submission. We are **extremely grateful** that you took the time to point out such a detailed list of errors (e.g., "Intervension", "moemory pool", "Editting", "effevtively", " $\gamma_2 0$ "). We **commit to correcting every error** you noted and will conduct a thorough, professional proofread of the entire manuscript for the final version.

Regarding your other technical concerns, we respond as follows:

Weaknesses

- **W2: (Fixed thresholds may fail under extreme distribution shifts)**
  - **Response:** This is a very insightful point. You are correct that the robustness of our static thresholds under extreme OOD streams has not been validated. We will **add this as a limitation and a clear direction for future work in our conclusion**, noting the potential for exploring adaptive thresholding techniques.
- **W3: (Editing heterogeneous samples may harm alignment)**
  - **Response:** This is an excellent suggestion to validate our distribution-aware design. We **commit to adding this new ablation study**. We will force the model to train on "heterogeneous batches" (i.e., disabling our similarity clustering) and compare its performance (especially OP and Locality) against our current method. We will add this result to our ablation studies (Figure 5).
- **W4: (REPAIR is similarity to RECIPE [1])**
  - **Response:** Thank you for pointing out this highly relevant work. We **commit to adding a detailed discussion of RECIPE** in our Related Work (Appendix B) and, more importantly, we will **add RECIPE [1] as a key baseline for full experimental comparison in Table 3** in the final version.
- **W5: (Insufficiently detailed process description)**
  - **Response:** You are correct; Figure 2 is a high-level diagram. We **commit to adding a more detailed step-by-step flowchart in the appendix** to clearly illustrate the full loop of learning, clustering, and editing, as you suggested.

Questions

- **Q1: (How is homogeneous batch similarity defined?)**
  - **Response:** We apologize for not making this explicit in Section 2.3. As detailed in **Algorithm 3 (line 6)**, we define similarity using the **cosine similarity** of the feature representations (o_i). We **commit to adding this definition to the main text of Section 2.3**.
- **Q2: (Could Closed Loop Feedback potentially harm the locality of existing knowledge?)**
  - **Response:** This is an excellent question about the core of our mechanism. We are happy to point out that our **Ablation Studies (Figure 5) already answer this**.
  - In **Figure 5(d) (** $N = 1000$ **)**, the **Locality (Loc.)** score of the full **REPAIR** (red line) is significantly *higher* than the **'w/o prune'** (orange-dashed line) variant.

- This demonstrates that our closed-loop feedback, far from harming locality, **actually *improves* it** by actively identifying and removing conflicting shards. We will explicitly highlight this key finding in Section 3.3.

We believe that by thoroughly correcting the presentation and by strengthening the soundness (with new experiments and clarifications), we can resolve all your concerns.