AlphaEdit$^+$: Model Editing in the Presence of Conflicting and Inconsistent Knowledge

Qing Liu[1,†], Jianhao Zhang[1,†], Ou Wu[1,*], Michael Ng[2], Yi Du[1]

[1]Hangzhou Institute for Advanced Study,
University of Chinese Academy of Sciences, Hangzhou, China,
[2]Faculty of Science, Hong Kong Baptist University, Hong Kong, China
{liuqing25, zhangjianhao24}@mails.ucas.ac.cn, wuou@ucas.ac.cn

Abstract

Knowledge editing is a crucial technique for daily updates in LLMs, requiring a balance between accurately modifying incorrect knowledge and preserving existing information. The recently proposed AlphaEdit method achieves competitive editing performance by updating parameters under null-space constraints. However, our theoretical analysis reveals that AlphaEdit struggles with high knowledge conflicts and inconsistencies during editing. To address this, we propose a new editing method AlphaEdit$^+$, featuring three key improvements: 1) relaxing null-space constraints by adding a matrix perturbation through optimization to resolve conflicts between new and preserved knowledge; 2) introducing a weighting scheme on previously updated knowledge constraints to mitigate conflicts between new and historical editing; 3) developing a value smoothing algorithm to resolve high knowledge inconsistencies. These enhancements collectively ensure robust editing while maintaining model coherence. Comprehensive experiments show that our approach AlphaEdit$^+$ not only resolves the brittleness of the original method on carefully constructed challenging datasets but also outperforms AlphaEdit on existing benchmark datasets. Code at: https://github.com/zjh-vinky/AlphaEdit_plus.

1 Introduction

Knowledge editing enables precise modifications to factual associations in LLMs, bypassing costly full retraining (Gupta et al., 2024; Zhang et al., 2024; Pan et al., 2025). Current approaches primarily comprise two paradigms: parameter-modifying techniques (e.g., UnKE (Deng et al., 2025), IFMET (Zhang et al., 2025), BaFT (Liu et al., 2025)) that directly edit critical model weights, and parameter-preserving methods (e.g., SERAC (Mitchell et al., 2022), MELO (Yu et al., 2024), postEdit (Song et al., 2024)) that store new knowledge in external modules or in-context editing via prompts (e.g., IKE (Zheng et al., 2023)). Although first-line techniques are efficient, the core challenge is integrating new knowledge while preserving existing capabilities, a balance difficult to achieve due to parameter shifting and limited effects on stored information. Recently, AlphaEdit (Fang et al., 2025) has emerged as an effective solution, minimizing retained knowledge interference via null-space projection for edits.

Despite significant advancements, existing knowledge editing studies critically overlook multifaceted conflicts between newly introduced knowledge, preserved foundational knowledge, and previously edited facts. In addition, severe inconsistencies may occur when edited knowledge diverges significantly from the model's existing parametric knowledge, manifested as logical contradictions (e.g., 'Paris is the capital of France' vs. new edit 'Roma is the capital of France') or cascading inference errors. Through mathematical analysis of AlphaEdit, we further characterize how these discrepancies destabilize its projection mechanism: highly conflicting between new edits and preserved knowledge spaces or pre-edits ultimately causing edit failure; and high levels of inconsistency exert significant adverse effects even on knowledge that is relatively easy to edit. Limited prior work has acknowledged these conflicts and inconsistencies, but a systematic investigation and comprehensive solution remain unexplored.

In this study, we quantify knowledge conflict and inconsistency for knowledge editing. We then propose AlphaEdit$^+$, an approach that resolves these conflicts and inconsistencies with three improvements. First, we introduce a perturbation matrix into the null-space matrix to alleviate its constraints; this perturbation matrix is added as an optimization variable within the objective func-

[†]Equal contributions.
[*]Corresponding Author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 14809–14835

tion to resolve conflicts arising between new edits and existing knowledge. Additionally, we incorporate a conflict-based weighting factor for previous edits' residuals, reducing effects of conflicting knowledge. Finally, for highly inconsistent edits, we adopt a progressive smoothing strategy for objectives, facilitating incremental updates to model parameters that approach the desired editing goal.

To investigate limitations of existing methods, we constructed three challenging datasets (AlphaSet1, AlphaSet2, AlphaSet3) targeting high-conflict and inconsistency scenarios, collectively called AlphaSet. Experiments show that existing methods suffer notable declines in editing scores under these settings, whereas our approach demonstrates consistently superior performance and robustness. These results highlight that, with conflict and/or inconsistency, our method achieves progressively larger advantages over current SOTA techniques, especially AlphaEdit.

Our main contributions are:

- Building upon the AlphaEdit framework, we establish a pioneering mathematical systematization of knowledge conflicts and inconsistencies in model editing, providing formal quantification and analyzing their impacts.
- We then introduce `AlphaEdit`$^+$, a novel methodology incorporating three key enhancements: perturbation-based adaptive null-space matrix relaxation for conflict alleviation between new edits and model knowledge, conflict-aware weighting for pre-edit constraints, and objective-oriented smoothing for refining knowledge with inference inconsistencies.
- Through comprehensive comparative experiments accompanied by ablation studies and sensitive tests, we conclusively demonstrate the efficacy of our proposed approach.

2 Theoretical Analyses for AlphaEdit

2.1 Preliminaries for AlphaEdit

Let $\mathbf{K}_0$, $\mathbf{K}_1$, and $\mathbf{K}_p$ be the key sets of the preserved, the to-be-updated, and previously updated knowledge, respectively. Their value sets are denoted by $\mathbf{V}_0$, $\mathbf{V}_1$, and $\mathbf{V}_p$, respectively. Let $\mathbf{W}$ be the parameters to be updated. Model editing seeks a perturbation $\mathbf{\Delta}$ of $\mathbf{W}$ so that $(\mathbf{K}_1, \mathbf{V}_1)$ is correctly stored while $(\mathbf{K}_0, \mathbf{V}_0)$ and $(\mathbf{K}_p, \mathbf{V}_p)$ are kept. Several classical methods, such as ROME (Meng et al., 2022) and AnyEdit (Jiang et al., 2025), have been developed based on this mechanism. Recently, AlphaEdit (Fang et al., 2025) imposes a *null-space* constraint on the edit matrix $\mathbf{\Delta}$. Concretely, let $\mathbf{P}$ be the orthogonal projector onto the left null space of $\mathbf{K}_0$, constructed via the SVD of $\mathbf{K}_0\mathbf{K}_0^\top$ such that $\mathbf{P} = \widehat{\mathbf{U}}\widehat{\mathbf{U}}^\top$ with $\widehat{\mathbf{U}}$ spanning the zero-eigen space. Consequently, the resulting objective[1] is:

$$\min_{\tilde{\mathbf{\Delta}}} \left\|(\mathbf{W}+\tilde{\mathbf{\Delta}}\mathbf{P})\mathbf{K}_1-\mathbf{V}_1\right\|_F^2+\left\|\tilde{\mathbf{\Delta}}\mathbf{P}\mathbf{K}_p\right\|_F^2+\lambda\left\|\tilde{\mathbf{\Delta}}\mathbf{P}\right\|_F^2, \quad (1)$$

where the second term penalizes interference with prior updated knowledge $\mathbf{K}_p$ (sequential editing), and the third term is a Tikhonov regularizer. Let $\mathbf{R} \triangleq \mathbf{V}_1 - \mathbf{W}\mathbf{K}_1$. Eq. 1 is a convex linear least-squares problem in $\mathbf{\Delta}$ and admits a closed form (Lang, 2012). The solution is:

$$\mathbf{\Delta}^* = \tilde{\mathbf{\Delta}}\mathbf{P} = \mathbf{R}\,\mathbf{K}_1^\top\mathbf{P}\left(\mathbf{K}_p\mathbf{K}_p^\top\mathbf{P}+\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}+\lambda\mathbf{I}\right)^{-1}, \quad (2)$$

where λ is a hyperparameter. As $\mathbf{P}$ can be obtained in advance, computation is efficient. Experiments in (Fang et al., 2025) sufficiently validate the superior performances of AlphaEdit.

2.2 Defect Analysis of AlphaEdit

This subsection mathematically demonstrates that AlphaEdit fails or produces suboptimal edits when severe knowledge conflicts and inconsistencies exist. We formally define knowledge conflict as follows: given a target edit $(\boldsymbol{k}, \boldsymbol{v})$, its conflict with a knowledge set $\mathbf{K}$ is quantified by:

$$s(\boldsymbol{k}, \mathbf{K}) = \max_{\boldsymbol{k}' \in \mathbf{K}} \frac{|\boldsymbol{k}^\top \boldsymbol{k}'|}{\|\boldsymbol{k}\|\,\|\boldsymbol{k}'\|} \in [0, 1]. \quad (3)$$

$\mathbf{K}$ can be $\mathbf{K}_0$ or $\mathbf{K}_p$. A high score reflects strong overlap with either preserved or previously updated knowledge, thereby indicates a high conflict between $\boldsymbol{k}$ and $\mathbf{K}$. In addition, knowledge inconsistency is mathematically captured by the residual norm $\|\boldsymbol{r}\| = \|\boldsymbol{v} - \mathbf{W}\boldsymbol{k}\|$, which measures the gap between the model's current knowledge (i.e., $\mathbf{W}\boldsymbol{k}$) and the target value. A large $\|\boldsymbol{r}\|$ implies a high inconsistency. In the following sections, we provide a detailed mathematical analysis of the impacts arising from three scenarios.

Conflict with Preserved Knowledge ($\mathbf{K}_0$). To simplify the analysis, we assume that $\mathbf{K}_1$ contains only a single knowledge tuple $(\boldsymbol{k}_1, \boldsymbol{v}_1)$ with $\|\boldsymbol{k}_1\| = 1$ requiring editing, and we disregard $\mathbf{K}_p$. In this scenario, the AlphaEdit solution simplifies

[1] In this study, we omit a minor weighting coefficient β on the prior-edit term and this simplification does not affect the theoretical analysis (see Appendix A.5).

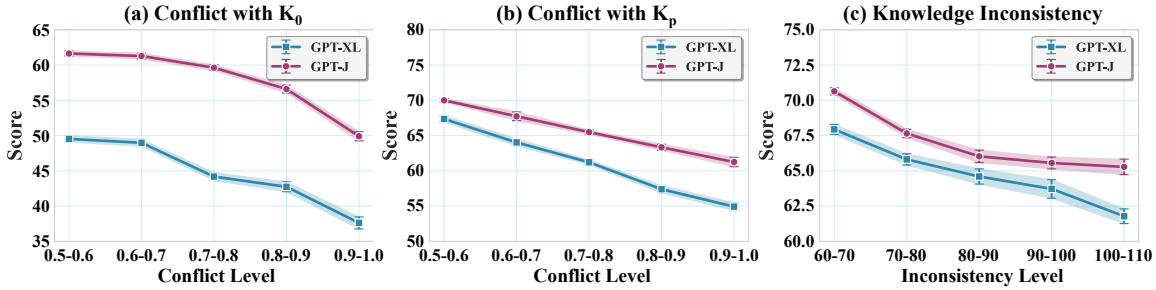

Figure 1: Edit scores versus conflict (similarity s) and inconsistency (residual $\|r\|$) for AlphaEdit on AlphaSet1, AlphaSet2, and AlphaSet3. The x-axis bins are left-open and right-closed intervals.

as $\mathbf{\Delta}^* = \mathbf{R}\,\boldsymbol{k}_1^\top \mathbf{P}\big(\boldsymbol{k}_1\boldsymbol{k}_1^\top \mathbf{P} + \lambda \mathbf{I}\big)^{-1}$. Through algebraic derivation, the solution simplifies to:

$$\mathbf{\Delta}^* = \frac{\lambda}{\lambda + \|\mathbf{P}\boldsymbol{k}_1\|^2}\,\boldsymbol{r}_1\,(\mathbf{P}\boldsymbol{k}_1)^\top. \tag{4}$$

We first establish the relationship between this solution and the conflict coefficient s. The vector $\boldsymbol{k}_1$ decomposes into components parallel and orthogonal to the column space of $\mathbf{K}_0$:

$$\boldsymbol{k}_1 = \boldsymbol{k}_{1\parallel} + \boldsymbol{k}_{1\perp}, \quad \begin{cases} \boldsymbol{k}_{1\parallel} \in \mathrm{col}(\mathbf{K}_0) \\ \boldsymbol{k}_{1\perp} \in \mathrm{null}(\mathbf{K}_0^\top) \end{cases}. \tag{5}$$

After projection, $\mathbf{P}\boldsymbol{k}_1 = \boldsymbol{k}_{1\perp}$ (since $\mathbf{P}$ projects onto the nullspace). Given that $\|\boldsymbol{k}_1\| = 1$, s is related to the norms as:

$$\|\boldsymbol{k}_{1\parallel}\|^2 \approx s^2, \quad \|\boldsymbol{k}_{1\perp}\|^2 = \|\boldsymbol{k}_1\|^2 - \|\boldsymbol{k}_{1\parallel}\|^2 = 1 - s^2. \tag{6}$$

Therefore, $\|\mathbf{P}\boldsymbol{k}_1\| = \|\boldsymbol{k}_{1\perp}\| = \sqrt{1-s^2}$. Then we have:$\mathbf{\Delta}^* = \frac{\lambda}{\lambda+1-s^2}\,\boldsymbol{r}_1\,(\mathbf{P}\boldsymbol{k}_1)^\top$. Accordingly, we have the following conclusion.

Lemma 1 *The residual for $\boldsymbol{k}_1$ after editing is:*

$$\|(\mathbf{W}+\mathbf{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1\| = \tfrac{\lambda}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}\|\boldsymbol{r}_1\| = \tfrac{\lambda}{\lambda+1-s^2}\|\boldsymbol{r}_1\|.$$

The proof of algebraic derivation and Lemma 1 is in the Appendix A.1. In the experiments, $\lambda \approx 0.1N$, where N is the number of new edits. Therefore in this theoretical case, λ can be considered as 0.1 (only one edit $(\boldsymbol{k}_1, \boldsymbol{v}_1)$). This lemma reveals that knowledge conflicts adversely affect editing success: smaller s yields better edits (this conclusion well explains the observation of (Duan et al., 2025), while $s \to 1$ results in complete edit failure (zero residual change). Our theoretical conclusion is further validated through real editing tasks. We construct three datasets(AlphaSet1, AlphaSet2 and AlphaSet3) to evaluate high conflicts with $\mathbf{K}_0$, high conflicts with $\mathbf{K}_p$, and high inconsistency, respectively. Using AlphaEdit, we analyze how editing performance relates to the degree of conflict or inconsistency; datasets details are in Section 4.1. As shown in Fig. 1(a), editing scores decrease consistently for both models as $\mathbf{K}_0$ conflict rises in AlphaSet1, with an average reduction of 11.82% from low to high conflict conditions.

Conflict with Prior Edits ($\mathbf{K}_p$). Consider a single new edit $(\boldsymbol{k}_1, \boldsymbol{v}_1)$ and one prior edit $\boldsymbol{k}_p$ with $\|\boldsymbol{k}_1\| = \|\boldsymbol{k}_p\| = 1$. Assume both are orthogonal to the preserved knowledge ($\mathbf{K}_0^\top \boldsymbol{k}_1 = \mathbf{K}_0^\top \boldsymbol{k}_p = 0$), hence $\mathbf{P}\boldsymbol{k}_1 = \boldsymbol{k}_1$ and $\mathbf{P}\boldsymbol{k}_p = \boldsymbol{k}_p$. Therefore, $s = \|\boldsymbol{k}_1^\top \boldsymbol{k}_p\| \in [0,1]$ and $\|\boldsymbol{r}_1\| = \|\boldsymbol{v}_1 - \mathbf{W}\boldsymbol{k}_1\|$ denote the conflict and the inconsistency degrees, respectively. Restricting Eq. 1 to the subspace $\mathrm{span}\boldsymbol{k}_1, \boldsymbol{k}_p$ and solving yields the closed-form update (detailed in the Appendix A.2):

$$\mathbf{\Delta}^* = \boldsymbol{r}_1\,\boldsymbol{k}_1^\top\big(\lambda \mathbf{I} + \boldsymbol{k}_1\boldsymbol{k}_1^\top + \boldsymbol{k}_p\boldsymbol{k}_p^\top\big)^{-1} = \boldsymbol{r}_1\,\tfrac{(\lambda+1)\,\boldsymbol{k}_1^\top - s\,\boldsymbol{k}_p^\top}{(\lambda+1)^2 - s^2}.$$

Consequently, we obtain the following.

Lemma 2 *The residual for the new edit satisfies:*

$$\big\|(\mathbf{W}+\mathbf{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1\big\| = \frac{\lambda(\lambda+1)}{(\lambda+1)^2 - s^2}\,\|\boldsymbol{r}_1\|.$$

When $s \to 0$ (no conflict with the prior edit), the factor reduces to $\frac{\lambda}{\lambda+1}$ (in this theoretical case, λ can be considered as 0.2), recovering the single-edit case. As $|s| \to 1$, $(\lambda+1)^2 - s^2$ shrinks and the residual increases, indicating stronger interference from prior edits. Fig. 1(b) shows on AlphaSet2, editing scores decrease by an average of 10.59% across both models as $\mathbf{K}_p$ conflict levels increase.

Knowledge Inconsistency ($\mathbf{W}\boldsymbol{k}_1$ vs. $\boldsymbol{v}_1$). We consider two edits to be updated where $\mathbf{K}_1 = [\boldsymbol{k}_{1,1}, \boldsymbol{k}_{1,2}]$, $\mathbf{V}_1 = [\boldsymbol{v}_{1,1}, \boldsymbol{v}_{1,2}]$, and $\mathbf{R} = \mathbf{V}_1 - \mathbf{W}\mathbf{K}_1 = [\boldsymbol{r}_{1,1}, \boldsymbol{r}_{1,2}]$. We disregard $\mathbf{K}_p$ and assume only the magnitude relation $\|\boldsymbol{r}_{1,2}\| = \rho\,\|\boldsymbol{r}_{1,1}\|$ with $\rho \geq 1$. Define the projection Gram matrix $\mathbf{G} \triangleq \mathbf{K}_1^\top \mathbf{P}\mathbf{K}_1$, which encodes the pairwise inner products of the projected keys and thereby collapses the high-dimensional problem onto $\mathrm{span}\{\mathbf{P}\boldsymbol{k}_{1,1}, \mathbf{P}\boldsymbol{k}_{1,2}\}$, enabling closed-form residuals via $(\mathbf{G} + \lambda \mathbf{I})^{-1}$.

$$\mathbf{G} \triangleq \mathbf{K}_1^\top \mathbf{P}\mathbf{K}_1 = \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix} \geq 0, \quad g_{ij} = \boldsymbol{k}_{1,i}^\top \mathbf{P}\boldsymbol{k}_{1,j},$$

$$D_\lambda = (\lambda + g_{11})(\lambda + g_{22}) - g_{12}^2 > 0.$$

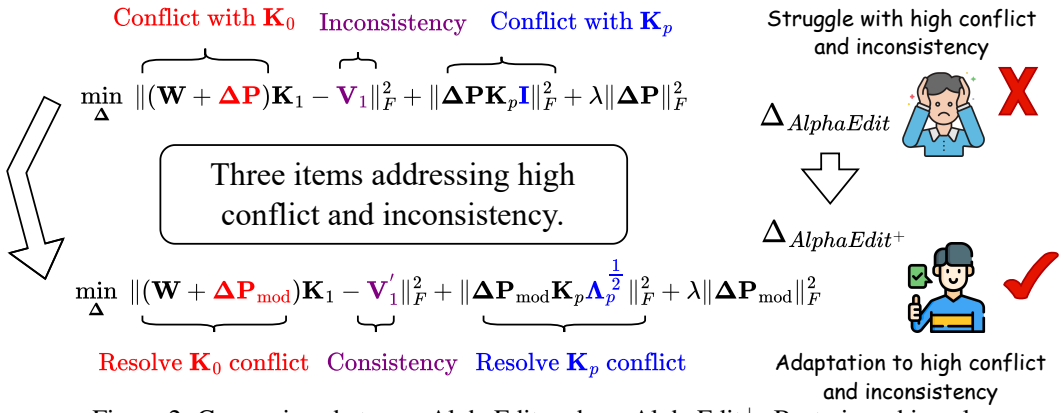

Figure 2: Comparison between AlphaEdit and our AlphaEdit$^+$. Best viewed in color.

Solving the ridge objective restricted to the projected subspace gives the residual matrix identity:

$$\mathbf{E} \triangleq (\mathbf{W}+\mathbf{\Delta}^*)\mathbf{K}_1 - \mathbf{V}_1 = -\lambda\mathbf{R}(\mathbf{G}+\lambda\mathbf{I})^{-1},$$

where

$$(\mathbf{G}+\lambda\mathbf{I})^{-1} = \tfrac{1}{D_\lambda}\begin{bmatrix}\lambda+g_{22} & -g_{12}\\ -g_{12} & \lambda+g_{11}\end{bmatrix}.$$

Let $\mathbf{E} = [\boldsymbol{e}_{1,1}, \boldsymbol{e}_{1,2}]$ denote the post-edit residual columns.

Lemma 3 *If $\|\boldsymbol{r}_{1,2}\| = \rho\|\boldsymbol{r}_{1,1}\|$ with $\rho \geq 1$, the post-edit residual for the new edit satisfies:*

$$\|\boldsymbol{e}_{1,1}\| \leq \frac{\lambda}{D_\lambda}\Big((\lambda+g_{22}) + \rho|g_{12}|\Big)\|\boldsymbol{r}_{1,1}\|,$$
$$\|\boldsymbol{e}_{1,2}\| \leq \frac{\lambda}{D_\lambda}\Big(|g_{12}| + \rho(\lambda+g_{11})\Big)\|\boldsymbol{r}_{1,1}\|.$$

The upper bound for the second key grows *linearly* with ρ at slope $\frac{\lambda}{D_\lambda}(\lambda+g_{11})$, i.e., a larger initial residual on the second edit inevitably raises its post-edit residual bound. The first key's upper bound also increases linearly with ρ at slope $\frac{\lambda}{D_\lambda}|g_{12}|$, reflecting cross-key coupling via g_{12}. Fig. 1(c) shows that on AlphaSet3, editing scores decline with increasing inconsistency, averaging a 5.76% reduction. Thus, editing algorithms must consider highly inconsistent data's harm.

3 Methodology

This section describes our proposed new method AlphaEdit$^+$. First, we establish an iterative optimization objective to resolve the aforementioned knowledge conflicts and inconsistencies, along with an iterative solution framework and detailed algorithmic implementation.

3.1 A Unified Optimization Objective

As shown in Lemma 1, when the updated knowledge $\mathbf{K}_1$ is in high conflict with $\mathbf{K}_0$, the projected norm $\|\mathbf{P}\boldsymbol{k}_1\|$ approaches zero, leading to poor performance. To address this, one may either modify $\mathbf{K}_0$-which requires recomputing the SVD and incurs high computational cost-or adjust $\mathbf{K}_1$ in a semantically invariant way to reduce conflict. We propose a novel alternative by introducing a perturbation to $\mathbf{P}$. As indicated in Lemma 2, high conflict between $\mathbf{K}_1$ and $\mathbf{K}_p$ also severely impairs editing efficacy. To mitigate this, one can modify either $\mathbf{K}_1$ or $\mathbf{K}_p$ without altering their semantics to reduce conflict. This work prioritizes modifying $\mathbf{K}_1$ to implicitly reduce the influence of $\mathbf{K}_p$. Furthermore, Lemma 3 demonstrates that the presence of hard samples in the dataset can substantially affect overall performance. Motivated by curriculum learning, we adopt a progressive strategy that smooths such samples to adjust their difficulty. Building on these insights, we formulate the optimization objective:

$$\begin{aligned}&\min_{\tilde{\mathbf{\Delta}},\tilde{\mathbf{P}}} \|(\mathbf{W}+\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}})\mathbf{K}_1 - \mathbf{V}_1^t\|_F^2 + \|\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}\mathbf{K}_p\mathbf{\Lambda}_p^{\frac{1}{2}}\|_F^2 \\ &+\lambda\|\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}\|_F^2, \text{s.t}\mathbf{P}_{\text{mod}} = \mathbf{P}+\tilde{\mathbf{P}}, \mathbf{P}\perp\tilde{\mathbf{P}}, \text{min rank}(\tilde{\mathbf{P}}).\end{aligned} \tag{7}$$

where $\mathbf{\Lambda}_p = \text{diag}(1-|s(\boldsymbol{k}_j, \mathbf{K}_1)|)$, $\boldsymbol{k}_j \in \mathbf{K}_p$ and λ is a hyperparameter; $\mathbf{V}_1^t$ denotes the smoothed targets[2] at iteration t:

$$\boldsymbol{v}_{1,i}^t = \boldsymbol{v}_{1,i} + \beta_i(t)(\boldsymbol{v}_{0,i} - \boldsymbol{v}_{1,i}), \beta_i(t) = \begin{cases}0, & \text{if}\|\boldsymbol{r}_i\| \leq \tau_r,\\ \dfrac{T-t}{2T}, & \text{otherwise},\end{cases}$$

with $\boldsymbol{r}_i = \boldsymbol{v}_{1,i} - \mathbf{W}\boldsymbol{k}_i$, current iteration t, threshold τ_r, smoothing factor $\boldsymbol{\beta}$, and total iterations T. Eq. 7 yields our `AlphaEdit`$^+$. For AlphaEdit and AlphaEdit$^+$ compared in Fig. 2, we have:

Lemma 4 *If $\forall\boldsymbol{k}_i \in \mathbf{K}_1$, $s(\boldsymbol{k}_i, \mathbf{K}_0) \equiv 0$, $\|\boldsymbol{r}_i\| \leq \tau_r$, and $\forall\boldsymbol{k}_j \in \mathbf{K}_p, s(\boldsymbol{k}_j, \mathbf{K}_1) \equiv 0$, then AlphaEdit$^+$ is reduced to AlphaEdit.*

[2]In this study, the average of $\boldsymbol{v}_1$ and $\boldsymbol{v}_0$ is set as the initial smoothed target.

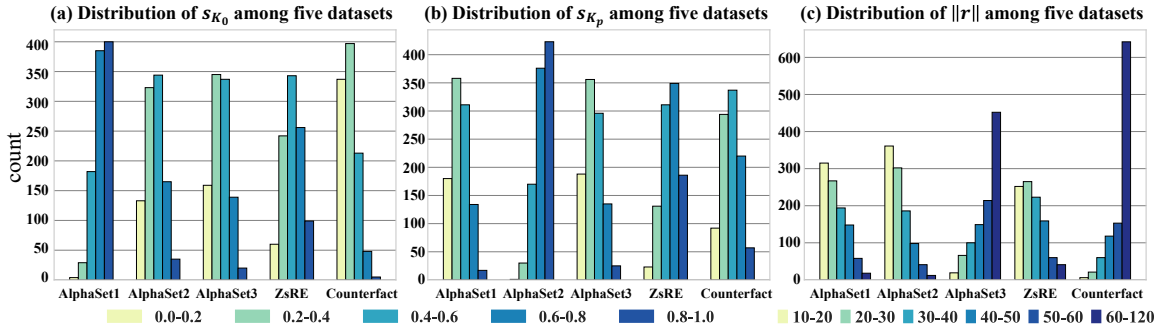

Figure 3: Distributions of $\mathbf{K}_0$ conflict, $\mathbf{K}_p$ conflict, and inconsistency across five datasets.

The proof appears in Appendix A.4. The next section describes how to iteratively solve Eq. 7.

3.2 The Solving for Eq. (7)

Our solving relies on a validation set to select the optimal solution among the candidates generated by iteratively optimizing Eq. 7. Given that existing mainstream methods do not require validation sets, our validation set construction adheres to the following principles: 1) Fairness: The dataset must not leak any test set information to ensure unbiased method comparison; 2) Simplicity: The construction process should be straightforward, as excessive complexity hinders practical application. The construction steps include: 1) sampling knowledge tuples for $\mathbf{K}_0$, $\mathbf{K}_1$, and $\mathbf{K}_p$, and 2) rewriting selected samples via LLMs. Details are in the experimental section and Appendix B.1.

Since Eq. 7 contains two coupled optimization variables, we propose a two-stage optimization scheme to ensure effective solutions. In the first stage ($t = 0$), we optimize $\tilde{\mathbf{P}}$ independently as its introduction is unrelated to $\mathbf{V}_1$. Given $\mathbf{P}_{\text{mod}} = \mathbf{P} + \tilde{\mathbf{P}}$, we have $\mathbf{\Delta}_+^{(0)} = \tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}$, expressed as:

$$\mathbf{R}^0\mathbf{K}_1^\top\mathbf{P}_{\text{mod}}(\mathbf{K}_p\mathbf{\Lambda}_p\mathbf{K}_p^\top\mathbf{P}_{\text{mod}}+\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}_{\text{mod}}+\lambda\mathbf{I})^{-1} \tag{8}$$

where $\mathbf{R}^0 = \mathbf{V}_1^0 - \mathbf{W}\mathbf{K}_1$. Assuming that $\mathbf{U}$ and $\mathbf{P}$ are given, let $\check{\mathbf{U}}$ be the set of sort eigenvectors with non-zero eigenvalues of $\mathbf{U}$ by ascending eigenvalue. We optimize $\tilde{\mathbf{P}}$ through a search-based approach, where at each search step we extract the eigenvector $\boldsymbol{u}_l$ corresponding to the current minimum eigenvalue from $\check{\mathbf{U}}$ (while removing it from $\check{\mathbf{U}}$ simultaneously), then update $\tilde{\mathbf{P}}$ as $\tilde{\mathbf{P}} = \tilde{\mathbf{P}} + \boldsymbol{u}_l\boldsymbol{u}_l^\top$. The modified $\mathbf{P}_{\text{mod}}$ is substituted into Eq. 8 to compute the new $\tilde{\mathbf{\Delta}}$, followed by evaluating the current value of $\left\|(\mathbf{W}+\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}})\mathbf{K}_1 - \mathbf{V}_1^0\right\|_F^2 + \left\|\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}\mathbf{K}_p\mathbf{\Lambda}_p^{\frac{1}{2}}\right\|_F^2 + \lambda\left\|\tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}\right\|_F^2$. If the total value is not reduced, then the search stops and the previous $\tilde{\mathbf{P}}$ is used as the solution.

Algorithm 1 AlphaEdit$^+$

Input: $\mathbf{K}_1$, $\mathbf{V}_1$, $\mathbf{K}_p$, $\mathbf{P}$, $\mathbf{U}$ and $\mathbf{\Sigma}$ from SVD($\mathbf{K}_0\mathbf{K}_0^\top$), $\mathbf{W}$, T, $\boldsymbol{\tau}_r$, ϵ, δ, λ, $\mathbf{\Lambda}_p = \mathbf{0}$, $\tilde{\mathbf{P}} = \mathbf{0}$, $\mathbf{\Delta}_+ = \mathbf{0}$.
Output: Final perturbation $\mathbf{\Delta}_+$.

```
Update Λ_p with Λ_p[j, j] = 1 − |s(k_j, K_1)|, k_j ∈ K_p.
Set Ǔ ← {u_l : σ_l > 0, σ_l ∈ Σ} sorted by increasing σ_l.
Compute R^(0), V_1^(0), β(0), and Eq. 7 objective at t = 0.
for each u_l ∈ Ǔ in order do
    P̃ ← P̃ + u_l u_l^⊤.
    Recompute the objective in Eq. 7 with t = 0.
    if value reduction < ε then
        P*_mod ← P + P̃ − u_l u_l^⊤; break.
    end if
end for
if ‖β(0)‖_1 = 0 then
    Δ_+ ← Eq. 8 using P*_mod;
    return Δ_+.
end if
Compute Δ_+^(0) with Eq. 8; evaluate val_score^(0).
for t = 1 to T do
    Compute R^(t), V_1^(t), β(t).
    Compute Δ_+^(t) with Eq. 9; evaluate val_score^(t).
    if |val_score^(t) − val_score^(t−1)| ≤ δ then break.
end for
return Δ_+^(t)
```

In the second stage ($t \geq 1$), we fixed $\mathbf{P}_{\text{mod}}^*$ and search the optimal $\tilde{\mathbf{\Delta}}$. In the t-th iteration, the current temporal optimal solution is as follows:

$$\mathbf{\Delta}_+^{(t)} = \boldsymbol{R}^t\mathbf{K}_1^\top\mathbf{P}_{\text{mod}}^*(\mathbf{K}_p\mathbf{\Lambda}_p\mathbf{K}_p^\top\mathbf{P}_{\text{mod}}^*+\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}_{\text{mod}}^*+\lambda\mathbf{I})^{-1}, \tag{9}$$

where $\mathbf{R}^t = \mathbf{V}_1^t - \mathbf{W}\mathbf{K}_1$ and $\mathbf{\Delta}_+^{(t)} = \tilde{\mathbf{\Delta}}\mathbf{P}_{\text{mod}}^*$. For this temporary optimal solution, we evaluate it using the previously constructed validation set and record the performance score. The optimization stops when the performance score no longer increases or when $t > T$.

The steps of the entire solving procedure are presented in Algorithm 1. Compared to AlphaEdit, the overall computational overhead mainly comes from repeated evaluations of Eqs. 8 and 9. Since only the term involving $\mathbf{R}^t$ changes in Eq. 9, other components can be reused to accelerate the process. Overall, the additional computational cost remains moderate, as confirmed experimentally,

Dataset	Method	GPT2-XL (1.5B)				GPT-J (6B)				LLaMA3 (8B)			
		Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑
AlphaSet1	MEMIT	70.66	56.44	22.22	49.77	98.95	90.19	27.10	72.08	83.18	81.32	30.92	65.14
$\text{avg}(s_{K_0}) = 0.79$	RECT	49.76	42.57	22.35	38.23	87.98	69.05	26.80	61.28	72.56	63.02	30.77	55.45
$\text{avg}(s_{K_p}) = 0.38$	PRUNE	92.50	82.54	23.77	66.27	96.91	91.13	27.76	71.93	91.21	83.26	31.54	68.67
$\text{avg}(\|r\|) = 17.67$	KDE	89.67	85.02	22.00	65.56	90.74	89.60	25.83	68.72	86.53	80.42	29.81	65.59
	BLUE	91.95	86.83	**24.33**	67.70	93.02	91.41	**28.16**	70.86	88.81	82.23	**32.14**	67.73
	AlphaEdit	78.67	63.45	22.64	54.92	97.24	82.33	27.15	68.91	87.75	80.85	30.97	66.52
	AlphaEdit$^+$	**98.69**	**87.98**	23.79	**70.15**	**99.76**	**92.56**	27.62	**73.31**	**95.55**	**83.38**	31.60	**70.18**
AlphaSet2	MEMIT	75.78	63.40	24.18	54.45	85.19	78.31	27.43	63.64	81.91	75.34	32.46	63.24
$\text{avg}(s_{K_0}) = 0.42$	RECT	65.91	52.12	24.20	47.41	82.08	68.60	27.13	59.27	77.27	66.32	32.45	58.68
$\text{avg}(s_{K_p}) = 0.81$	PRUNE	78.62	70.76	24.26	57.88	78.25	71.83	27.34	59.14	82.25	69.72	31.65	61.21
$\text{avg}(\|r\|) = 12.54$	KDE	92.46	75.72	24.32	64.17	94.78	79.03	27.24	67.02	88.23	78.54	32.22	66.33
	BLUE	90.34	75.01	24.24	63.20	92.66	78.32	27.16	66.05	86.11	77.83	32.14	65.36
	AlphaEdit	88.27	75.87	24.38	62.84	96.02	79.93	26.99	67.65	84.38	75.24	32.54	64.05
	AlphaEdit$^+$	**96.77**	**77.12**	**24.72**	**66.20**	**99.09**	**80.43**	**27.64**	**69.05**	**92.54**	**79.94**	**32.62**	**68.37**
AlphaSet3	MEMIT	58.38	47.28	22.41	42.69	93.53	83.87	25.72	67.71	90.84	71.96	31.64	64.81
$\text{avg}(s_{K_0}) = 0.39$	RECT	39.53	32.62	21.24	31.13	81.17	61.98	25.41	56.19	71.49	55.51	31.57	52.86
$\text{avg}(s_{K_p}) = 0.36$	PRUNE	80.12	72.81	**22.42**	58.45	93.53	83.87	**25.75**	67.72	90.04	**77.34**	30.23	65.87
$\text{avg}(\|r\|) = 57.29$	KDE	81.45	75.24	20.26	58.98	91.35	83.02	23.53	65.97	90.52	71.21	29.53	63.75
	BLUE	83.74	77.45	20.97	60.72	93.64	85.23	24.24	67.70	92.81	73.42	30.24	65.49
	AlphaEdit	79.10	74.58	20.31	58.00	92.94	84.65	24.84	67.48	90.86	72.15	**31.68**	64.90
	AlphaEdit$^+$	**85.79**	**78.19**	22.41	**62.13**	**95.69**	**85.97**	25.68	**69.11**	**94.86**	74.16	**31.68**	**66.90**
ZsRE	MEMIT	87.22	83.21	26.41	65.61	98.97	98.54	27.73	75.08	89.67	87.04	30.96	69.22
$\text{avg}(s_{K_0}) = 0.53$	RECT	77.90	69.85	25.60	57.78	96.95	93.07	28.04	72.69	86.32	80.14	30.85	65.77
$\text{avg}(s_{K_p}) = 0.62$	PRUNE	84.85	82.09	27.47	64.80	96.04	95.84	**34.17**	75.35	82.10	87.57	30.23	66.63
$\text{avg}(\|r\|) = 21.6$	KDE	96.31	92.81	27.22	72.11	97.16	97.26	32.87	75.76	92.61	90.53	32.92	72.02
	BLUE	98.21	93.42	26.53	72.72	99.71	97.87	32.18	76.59	93.12	91.14	32.23	72.16
	AlphaEdit	97.85	93.26	26.56	72.56	99.66	**99.15**	27.70	75.50	91.81	90.91	32.35	71.69
	AlphaEdit$^+$	**98.97**	**94.68**	**27.77**	**73.81**	**99.82**	99.13	33.42	**77.46**	**95.27**	**92.40**	**33.47**	**73.71**
Counterfact	MEMIT	87.00	48.75	**12.95**	49.57	92.50	67.50	14.35	58.11	87.75	68.90	23.20	59.95
$\text{avg}(s_{K_0}) = 0.23$	RECT	67.00	31.75	12.00	36.92	96.00	48.75	14.50	53.08	69.75	63.25	23.65	52.22
$\text{avg}(s_{K_p}) = 0.47$	PRUNE	90.00	69.25	10.40	56.55	93.32	72.50	12.65	59.49	79.50	75.12	21.20	58.61
$\text{avg}(\|r\|) = 88.54$	KDE	94.39	69.74	11.13	58.42	95.33	71.90	13.18	60.14	93.12	91.23	24.14	69.50
	BLUE	95.58	70.45	11.51	59.18	96.52	72.61	13.56	60.90	94.31	91.94	24.52	70.26
	AlphaEdit	95.50	66.75	10.75	57.67	97.50	70.75	14.15	60.80	92.50	91.00	23.40	68.97
	AlphaEdit$^+$	**98.02**	**71.63**	12.64	**60.76**	**98.96**	**73.79**	**14.69**	**62.48**	**96.75**	**93.12**	**25.65**	**71.84**

Table 1: Mean conflict/inconsistency levels and overall results on five datasets (%). Best per block in bold.

with only a slight increase observed. Future work will explore efficient approximations for inverting the matrix in Eq. 8 to further enhance computational efficiency.

4 Experiments

We conduct systematic experiments to answer three core questions: (1) robustness & efficacy: whether AlphaEdit$^+$ reduces editing failures under high conflict with preserved knowledge and prior edits, as well as inconsistency, and its runtime overhead relative to AlphaEdit; (2) general capability preservation: how well models edited by AlphaEdit$^+$ retain downstream abilities after sequences of challenging edits; and (3) effects on hidden representations: whether the new components (projection perturbation, conflict-aware weighting, value smoothing) change representations of unedited knowledge.

4.1 Experiment Setup

Base LLMs & Baselines. Our experiments are conducted on three LLMs: GPT2-XL (1.5B) (Radford et al., 2019), GPT-J (6B) (Wang and Komatsuzaki, 2021) and LLaMA3 (8B) (Meta, 2024). We compare our method against several model editing baselines, including MEMIT (Meng et al., 2023), PRUNE (Ma et al., 2025), RECT (Gu et al., 2024), AlphaEdit$_{\text{BLUE}}$ (Li et al., 2025) (denoted as BLUE in our tables), KDE (Xu et al., 2025), and AlphaEdit (Fang et al., 2025). For the hyperparameter settings of the baseline methods, we used the original code from the respective papers. Unless otherwise noted, we use hyperparameters: shared settings $T = 10$, $\lambda = 10$, $\delta = 2 \times 10^{-4}$ for three models; and model-specific $\epsilon = 3.0 \times 10^{-4}$ for GPT-2-XL, 1.2×10^{-4} for GPT-J and 1.4×10^{-4} for LLaMA3 (Algorithm 1). Experiments were on a single NVIDIA A100 GPU (80 GB).

Datasets and Metrics. As noted in Section 2.2, we construct three datasets. First, we introduce a new dataset, **AlphaSet**, built upon the widely used model-editing benchmarks ZsRE (Levy et al., 2017) and Counterfact (Meng et al., 2022), and further augmented with samples derived from Wikipedia (Vrandečić and Krötzsch, 2014). AlphaSet consists of three subsets: conflict with preserved knowledge (AlphaSet1), conflict with prior

edits (AlphaSet2), and knowledge inconsistency (AlphaSet3), comprising a total of 3,500 examples. For experiments involving smoothing, we design dedicated validation sets to identify the overall optimal solution (Appendix B.1). Second, we retain the original ZsRE and Counterfact benchmarks as separate testbeds to verify effectiveness on standard datasets. Third, to assess downstream general abilities, we use six tasks: SST (Socher et al., 2013), MRPC (Dolan and Brockett, 2005), MMLU (Hendrycks et al., 2021), RTE (Bentivogli et al., 2009), CoLA (Warstadt et al., 2019), and NLI (Williams et al., 2018). To evaluate the editing performance, we utilize three classical metrics: Efficacy (indicating edit success), Generalization (indicating paraphrase success), and Specificity (indicating neighborhood preservation). These metrics are computed based on strict Top-1 accuracy in a consistent manner across all datasets (see Appendix B.2 for details), and their macro-average Score is calculated, where higher values signify superior performance. Additionally, we report the F1 scores on six downstream tasks to assess the retention of general capabilities.

4.2 Performance on five datasets

Prior to experimentation, we computed and analyzed the distributions of two conflict types and one inconsistency across all five datasets, each comprising 1,000 instances. As shown in Fig. 3, $\mathbf{K}_0$ conflicts are most pronounced in AlphaSet1, while $\mathbf{K}_p$ conflicts are most severe in AlphaSet2, compared to other datasets. Inconsistencies are extremely severe in Counterfact and second most severe in AlphaSet3. These observations confirm that our constructed datasets well reflect knowledge conflict and inconsistency characteristics.

To evaluate the efficacy and robustness of AlphaEdit$^+$, we conduct experiments on three challenging datasets and further assess it on ZsRE and Counterfact with 200 sequential edits. As our method targets sample-level inconsistent knowledge rather than globally large residuals, we set the inconsistency threshold τ_r to the top 20% of inconsistency levels in each dataset (more threshold results are provided in Appendix C.4).

As shown in Table 1, AlphaEdit$^+$ consistently and substantially outperforms AlphaEdit and other baselines on nearly all metrics in AlphaSet. Efficacy and Generalization rise by 7.06% and 5.63% on average across three backbone models, while Specificity is preserved or even improved for LLaMA3. On existing datasets, AlphaEdit$^+$ remains highly competitive: on ZsRE and Counterfact, it achieves the highest Efficacy and overall editing score. These gains are attributed to the moderate conflict and inconsistency levels in both datasets. Overall, AlphaEdit$^+$ improves the average editing score by 4.46% over AlphaEdit on our carefully constructed datasets and by a further 2.15% on standard benchmarks.

Dataset	AlphaEdit	AlphaEdit$^+$
AlphaSet1	121.34s	163.53s
AlphaSet2	112.42s	104.52s
AlphaSet3	204.43s	482.58s
ZsRE	423.85s	579.56s
Counterfact	443.96s	710.65s

Table 2: Comparison of runtimes on GPT2-XL.

As shown in Table 2, our method requires three times the runtime of AlphaEdit on AlphaSet3, mainly from smoothing iterations. However, absolute editing times remain low, making the overhead acceptable given the performance gains.

Our algorithm naturally incorporates an adaptive early stopping mechanism, which terminates the progressive smoothing loop when the validation score improvement falls below a specific threshold δ. As shown in Table 3, moderately relaxing this threshold (e.g., to $\delta = 5\text{e}{-4}$) significantly reduces the runtime by approximately 28% while still maintaining a dominant performance advantage over the baseline. Furthermore, we explored parallelizing the iterative smoothing process. Due to current memory constraints for storing weight parameters on our hardware, we implemented a lightweight version processing five concurrent smoothing updates. Despite this limitation, we observed a substantial speedup, reducing the runtime to 578.50s.

Method / Setting	Runtime (s)	Score
AlphaEdit (Baseline)	418.20	67.67
AlphaEdit$^+$ ($\delta = 2\text{e}{-4}$)	956.62	**70.37**
AlphaEdit$^+$ ($\delta = 5\text{e}{-4}$)	685.40	69.91
AlphaEdit$^+$ ($\delta = 1\text{e}{-3}$)	615.21	68.95
AlphaEdit$^+$ (Parallel, $\delta = 2\text{e}{-4}$)	578.50	69.16

Table 3: Sensitivity of Runtime and Performance to δ and Parallelization on AlphaSet using GPT2-XL.

These results demonstrate that through basic engineering optimizations (adaptive early stopping and parallelization), the computational cost of AlphaEdit$^+$ can be strictly controlled and brought

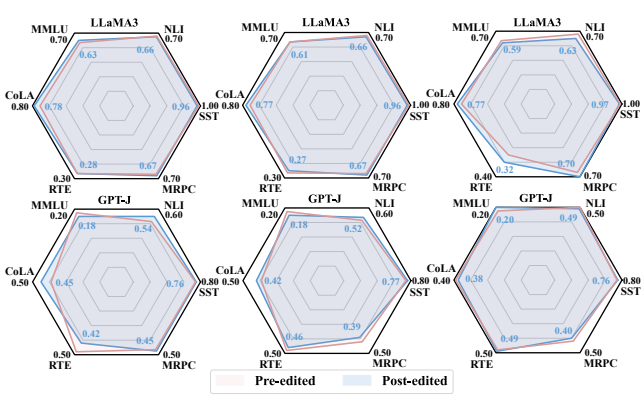

Figure 4: The effect of AlphaEdit$^+$ on F1 scores of two models on AlphaSet, ZsRE, CounterFact. Best viewed in color.

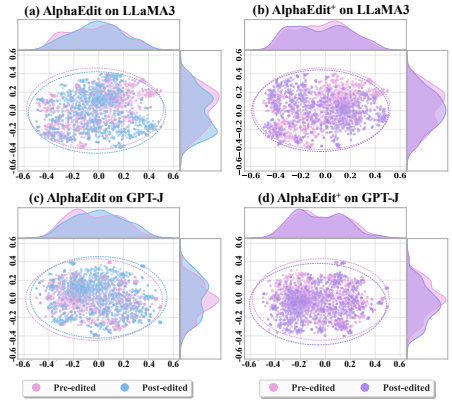

Figure 5: The distribution of hidden representations after dimensionality reduction.

to a highly acceptable level for practical use, without compromising its superior editing robustness.

To further evaluate the intrinsic knowledge of edited LLMs, we conduct General Capability Tests on six tasks from the GLUE benchmark (Wang et al., 2018). Specifically, we assess AlphaEdit$^+$ on AlphaSet, ZsRE, and Counterfact to compare the capabilities of two models before and after editing. The evaluation, summarized in Fig. 4, follows 2,000 sequential edits. The results show minimal impact on the models' general capabilities: except for a slight drop of 0.08 on RTE in GPT-J, original capabilities are well preserved across all settings. Overall, these results confirm that the three modules maintain general capabilities under large-scale sequential editing.

4.3 Hidden State Analysis & Ablation Tests

For analysis the hidden state we randomly select 1,000 factual prompts and extract the hidden representations within pre-edited LLMs. Subsequently, we performed AlphaSet, ZsRE and Counterfact on the LLMs and recomputed these hidden representation. Finally, we used t-SNE (Maaten and Hinton, 2008) to visualize the hidden representation before and after editing. Fig. 5 exhibits them and their marginal distribution curves. AlphaEdit$^+$ maintains consistency in hidden representations with AlphaEdit after editing. Specifically, in LLMs edited by AlphaEdit$^+$, the hidden representations align closely with the original distribution across two base models. This stability shows the three components introduced in AlphaEdit$^+$, projection perturbation, conflict-aware weighting, and smoothing, prevent distributional shifts in the models hidden representations while mitigating overfitting.

Variant	GPT2-XL (1.5B)			GPT-J (6B)			LLaMA3 (8B)		
	Eff↑	Gen↑	Spe↑	Eff↑	Gen↑	Spe↑	Eff↑	Gen↑	Spe↑
AlphaEdit$^+$	**96.72**	**85.42**	**28.97**	**98.54**	**89.73**	**30.73**	**96.97**	**82.10**	30.22
w/o $\tilde{\mathbf{P}}$	94.93	83.71	25.85	95.63	87.96	29.04	93.79	76.98	**32.60**
w/o $\mathbf{\Lambda}_p$	94.12	83.97	26.54	96.18	87.97	28.20	93.09	80.94	29.88
w/o Smoothing	93.51	81.38	25.46	95.36	86.19	29.17	90.50	78.30	30.78

Table 4: Ablation analysis of AlphaEdit$^+$ on AlphaSet for GPT2-XL, GPT-J and LLaMA3: optimal results per metric highlighted in **Bold**.

Beyond the overall improvements, we further conduct an ablation study to disentangle the contribution of each component. Table 4 reports ablations on AlphaSet, which includes conflicts and inconsistencies. Disabling each module leads to measurable drops: removing projection perturbation weakens efficacy and specificity, removing conflict-aware weighting reduces generalization, and discarding smoothing causes the largest decline in efficacy. These results confirm that the three components play complementary roles, and that full AlphaEdit$^+$ consistently achieves the most balanced performance across models.

5 Related Work

Knowledge Editing. Parameter-modifying knowledge editing enables efficient LLM factual updates via direct weight adjustments, avoiding costly retraining. Early approaches like ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023) prioritize accuracy but lack constraints, risking interference with existing knowledge. Subsequently, AlphaEdit (Fang et al., 2025) introduced null-space projection to confine updates to the orthogonal subspace of preserved knowledge. Complementary efforts include AnyEdit (Jiang et al., 2025), SIR (Wang et al., 2025a), and PRUNE (Ma et al., 2025), focusing on generalization and numerical stability. More recent work stress-tests editors in

sequential settings, where performance often degrades, including robustness failures under paraphrases and long-context queries (Sun et al., 2025; Hu et al., 2024; Yan et al., 2025). DeltaEdit (Cao et al., 2025) attributes such failures to accumulated superimposed noise and employs dynamic orthogonal constraints, while Wei et al. (2025) identify knowledge element overlap (KEO) as a source of conflict.

Learning Under Imperfect Data. Models are often trained or updated under imperfect supervision, such as noisy labels (Song et al., 2022) or intrinsically hard examples (Zhou et al., 2025), which has motivated two classic strategies: example weighting (Xie et al., 2025) and target/data smoothing (Rangwani et al., 2022). Weighting methods down-weight harmful signals; meta-reweighting learns per-example weights from a small clean set to improve robustness (Ren et al., 2018). Sequential smoothing and differencing denoise and stabilize time-series data, improving deep-learning forecasting (Livieris et al., 2021). In knowledge editing, recent studies have begun to investigate edit difficulty. SCIENCEMETER (Wang et al., 2025b) reports scientific frontier/ambiguous claims are prone to erroneous updates; SGR-Edit (Chen et al., 2025) shows short-answer edits overfit, while rationales generalize better. (Ge et al., 2024) used target perplexity to measure edit difficulty. Yet perplexity is static. We introduce a dynamic, optimization-aligned difficulty indicator, the editing residual, aligning with studies that use training losses as difficulty indicators.

6 Conclusion

We tackle an underexplored failure mode in knowledge editing: edits conflicting with prior knowledge or parametric beliefs, and show theoretically and empirically that AlphaEdit's null-space projection degrades in these regimes. We present AlphaEdit$^+$, addressing this via: (1) a learnable projection perturbation to relax constraints, (2) conflict-aware weighting to reduce interference, and (3) progressive target smoothing for large-residual edits. On GPT2-XL, GPT-J and LLaMA3, AlphaEdit$^+$ increases edit success and generalization while maintaining specificity and general capabilities with moderate overhead. Thus, principled control of projection geometry, edit weighting, and smoothing enables robust, minimally disruptive editing. Future work will investigate better difficulty measures, refine smoothing for high inconsistency, and explore transferring our approach to other parameter-modifying frameworks.

Limitations

Although AlphaEdit$^+$ demonstrates strong theoretical and empirical performance in complex editing scenarios, we acknowledge three limitations. (1) While AlphaSet captures diverse conflict patterns, its moderate scale and partially synthetic construction may not fully reflect real-world long-tail distributions. Future work will incorporate larger and more diverse natural corpora. (2) Our lightweight validation mechanism facilitates practical adoption but limits optimal solution selection. We plan to explore more principled criteria and richer natural supervision signals. (3) Although cross-term interactions amplify inconsistency (Lemma 3), AlphaEdit$^+$ addresses them indirectly via residual smoothing to avoid the high cost of explicit high-dimensional optimization. Developing efficient approximations for direct cross-term control remains an important direction.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 62476191 and 42550187. We would also like to thank the anonymous reviewers and the area chair for their constructive comments. We also acknowledge the computing resources provided by our research group.

Ethics Statement

This work aims to advance knowledge editing techniques in LLMs. Our proposed framework, AlphaEdit$^+$, is designed to improve robustness in the presence of conflicting and inconsistent knowledge without requiring full retraining. We emphasize that this research is intended solely for responsible scientific exploration, supporting safe, transparent, and efficient model maintenance. We do not attempt to use model editing techniques for generating harmful, biased, or misleading content. The datasets employed (ZsRE, Counterfact, and Wikipedia) are all publicly available, widely adopted in prior research, and contain no personally identifiable or sensitive information. The

methods and results presented in this paper are intended to promote reproducibility and enable critical evaluation by the research community, in line with ethical standards for AI research.

References

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. *TAC*, 7(8):1.

Ding Cao, Yuchen Cai, Rongxi Guo, Xuesong He, and Guiquan Liu. 2025. Deltaedit: Enhancing sequential editing in large language models by controlling superimposed noise. *arXiv preprint arXiv:2505.07899*.

Shigeng Chen, Linhao Luo, Zhangchi Qiu, Yanan Cao, Carl Yang, and Shirui Pan. 2025. Beyond memorization: A rigorous evaluation framework for medical knowledge editing. *arXiv preprint arXiv:2506.03490*.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298.

Jingcheng Deng, Zihao Wei, Liang Pang, Hanxing Ding, Huawei Shen, and Xueqi Cheng. 2025. Everything is editable: extend knowledge editing to unstructured data in large language models. In *The Thirteenth International Conference on Learning Representations*.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third international workshop on paraphrasing (IWP2005)*.

Zenghao Duan, Wenbin Duan, Zhiyi Yin, Yinghan Shen, Shaoling Jing, Jie Zhang, Huawei Shen, and Xueqi Cheng. 2025. Related knowledge perturbation matters: Rethinking multiple pieces of knowledge editing in same-subject. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 363–373.

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 *The Thirteenth International Conference on Learning Representations*.

Huaizhi Ge, Frank Rudzicz, and Zining Zhu. 2024. How well can knowledge edit methods edit perplexing knowledge? *arXiv preprint arXiv:2406.17253*.

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. 2024. Model editing harms general abilities of large language models: Regularization to the rescue. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16801–16819.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. 2024. Model editing at scale leads to gradual and catastrophic forgetting. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 15202–15232.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Chenhui Hu, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Wilke: Wise-layer knowledge editor for lifelong knowledge editing. *arXiv preprint arXiv:2402.10987*.

Houcheng Jiang, Junfeng Fang, Ningyu Zhang, Mingyang Wan, Guojun Ma, Xiang Wang, Xiangnan He, and Tat-Seng Chua. 2025. Anyedit: Edit any knowledge encoded in language models. In *Forty-second International Conference on Machine Learning*.

Serge Lang. 2012. *Introduction to linear algebra*. Springer Science & Business Media.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *21st Conference on Computational Natural Language Learning*, pages 333–342. Association for Computational Linguistics.

Xiaopeng Li, Shangwen Wang, Shasha Li, Shezheng Song, Bin Ji, Ma Jun, and Jie Yu. 2025. Rethinking residual distribution in locate-then-edit model editing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Tianci Liu, Ruirui Li, Yunzhe Qi, Hui Liu, Xianfeng Tang, Tianqi Zheng, Qingyu Yin, Monica Xiao Cheng, Jun Huan, Haoyu Wang, and Jing Gao. 2025. Unlocking efficient, scalable, and continual knowledge editing with basis-level representation fine-tuning. In *The Thirteenth International Conference on Learning Representations*.

Ioannis E Livieris, Stavros Stavroyiannis, Lazaros Iliadis, and Panagiotis Pintelas. 2021. Smoothing and stationarity enforcement framework for deep learning time-series forecasting. *Neural Computing and Applications*, 33(20):14021–14035.

Jun-Yu Ma, Hong Wang, Hao-Xiang Xu, Zhen-Hua Ling, and Jia-Chen Gu. 2025. Perturbation-restrained sequential model editing. In *The Thirteenth International Conference on Learning Representations*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *International Conference on Learning Representations*.

Meta. 2024. Llama 3. Large language model release.

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.

Haowen Pan, Xiaozhi Wang, Yixin Cao, Zenglin Shi, Xun Yang, Juanzi Li, and Meng Wang. 2025. Precise localization of memories: A fine-grained neuron-level knowledge editing technique for llms. In *The Thirteenth International Conference on Learning Representations*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Harsh Rangwani, Sumukh K Aithal, Mayank Mishra, Arihant Jain, and Venkatesh Babu Radhakrishnan. 2022. A closer look at smoothness in domain adversarial training. In *International conference on machine learning*, pages 18378–18399. PMLR.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pages 4334–4343. PMLR.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 34(11):8135–8153.

Xiaoshuai Song, Zhengyang Wang, Keqing He, Guanting Dong, Yutao Mou, Jinxu Zhao, and Weiran Xu. 2024. Knowledge editing on black-box large language models. *CoRR*.

Wei Sun, Tingyu Qu, Mingxiao Li, Jesse Davis, and Marie Francine Moens. 2025. Mitigating negative interference in multilingual knowledge editing through null-space constraints. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8796–8810.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, page 353. Association for Computational Linguistics.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 billion parameter autoregressive language model.

Jianchen Wang, Zhouhong Gu, Xiaoxuan Zhu, Lin Zhang, Haoning Ye, Zhuozhi Xiong, Sihang Jiang, Hongwei Feng, and Yanghua Xiao. 2025a. The missing piece in model editing: A deep dive into the hidden damage brought by model editing. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Yike Wang, Shangbin Feng, Yulia Tsvetkov, and Hannaneh Hajishirzi. 2025b. Sciencemeter: Tracking scientific knowledge updates in language models. *arXiv preprint arXiv:2505.24302*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Yifan Wei, Xiaoyan Yu, Ran Song, Hao Peng, and Angsheng Li. 2025. Setke: Knowledge editing for knowledge elements overlap. *arXiv preprint arXiv:2504.20972*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1112–1122.

Amber Xie, Rahul Chand, Dorsa Sadigh, and Joey Hejna. 2025. Data retrieval with importance weights for few-shot imitation learning. In *9th Annual Conference on Robot Learning*.

Haoyu Xu, Pengxiang Lan, Enneng Yang, Guibing Guo, Jianzhe Zhao, Linying Jiang, and Xingwei Wang. 2025. Knowledge decoupling via orthogonal projection for lifelong editing of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13194–13213.

Jianhao Yan, Futing Wang, Yun Luo, Yafu Li, and Yue Zhang. 2025. Keys to robust edits: from theoretical insights to practical advances. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22545–22560.

Lang Yu, Qin Chen, Jie Zhou, and Liang He. 2024. Melo: Enhancing model editing with neuron-indexed dynamic lora. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19449–19457.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, and 1 others. 2024. A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.

Zhuoran Zhang, Yongxiang Li, Zijian Kan, Keyuan Cheng, Lijie Hu, and Di Wang. 2025. Locate-then-edit for multi-hop factual recall under knowledge editing. In *Forty-second International Conference on Machine Learning*.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. 2023. Can we edit factual knowledge by in-context learning? In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Chao Zhou, Cheng Qiu, Lizhen Liang, and Daniel E Acuna. 2025. Paraphrase identification with deep learning: A review of datasets and methods. *IEEE Access*.

A Theoretical Analysis

This section provides detailed mathematical derivations for the results and lemmas in Sec. 2.2. We strictly follow the notations used in the main text.

A.1 Conflict with Preserved Knowledge ($\mathbf{K}_0$)

We consider the simplified case with a single edit $(\boldsymbol{k}_1, \boldsymbol{v}_1)$ while ignoring $\mathbf{K}_p$. In this situation, the AlphaEdit solution reduces from Eq. 2 to

$$\boldsymbol{\Delta}^* = \mathbf{R}\,\boldsymbol{k}_1^\top \mathbf{P}\,(\boldsymbol{k}_1\boldsymbol{k}_1^\top \mathbf{P} + \lambda \mathbf{I})^{-1}, \quad \mathbf{R} = \boldsymbol{r}_1. \tag{10}$$

Since $\boldsymbol{k}_1\boldsymbol{k}_1^\top\mathbf{P}$ has rank at most one, let us denote $\boldsymbol{u} = \mathbf{P}\boldsymbol{k}_1$. Then $\boldsymbol{k}_1\boldsymbol{k}_1^\top\mathbf{P} = \boldsymbol{k}_1\boldsymbol{u}^\top$, and the inverse becomes $(\lambda\mathbf{I}+\boldsymbol{k}_1\boldsymbol{u}^\top)^{-1}$. By the Sherman–Morrison formula,

$$(\lambda\mathbf{I} + \boldsymbol{k}_1\boldsymbol{u}^\top)^{-1} = \frac{1}{\lambda}\mathbf{I} - \frac{1}{\lambda^2}\,\frac{\boldsymbol{k}_1\boldsymbol{u}^\top}{1+\frac{1}{\lambda}\boldsymbol{u}^\top\boldsymbol{k}_1}. \tag{11}$$

Substituting this expression back into the solution yields

$$\boldsymbol{\Delta}^* = \boldsymbol{r}_1\,\boldsymbol{k}_1^\top\mathbf{P}\left(\frac{1}{\lambda}\mathbf{I} - \frac{1}{\lambda^2}\,\frac{\boldsymbol{k}_1\boldsymbol{u}^\top}{1+\frac{1}{\lambda}\boldsymbol{u}^\top\boldsymbol{k}_1}\right). \tag{12}$$

Since $\boldsymbol{k}_1^\top\mathbf{P} = \boldsymbol{u}^\top$, the first term becomes $\frac{1}{\lambda}\,\boldsymbol{r}_1\boldsymbol{u}^\top$, while the second term equals

$$-\boldsymbol{r}_1\frac{1}{\lambda^2}\cdot\frac{\boldsymbol{u}^\top\boldsymbol{k}_1}{1+\frac{1}{\lambda}\boldsymbol{u}^\top\boldsymbol{k}_1}\,\boldsymbol{u}^\top. \tag{13}$$

Factoring out $\boldsymbol{r}_1\boldsymbol{u}^\top$, the coefficient is

$$\frac{1}{\lambda} - \frac{1}{\lambda^2}\cdot\frac{\boldsymbol{u}^\top\boldsymbol{k}_1}{1+\frac{1}{\lambda}\boldsymbol{u}^\top\boldsymbol{k}_1}. \tag{14}$$

Since $\boldsymbol{u} = \mathbf{P}\boldsymbol{k}_1$ and $\mathbf{P}$ is a projection, we have $\boldsymbol{u}^\top\boldsymbol{k}_1 = \|\boldsymbol{u}\|^2$. Denoting $\|\boldsymbol{u}\|^2 = \|\mathbf{P}\boldsymbol{k}_1\|^2$, the coefficient simplifies to $\frac{1}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}$. Hence the closed-form solution becomes

$$\boldsymbol{\Delta}^* = \frac{1}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}\,\boldsymbol{r}_1\,\boldsymbol{u}^\top = \frac{\lambda}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}\,\boldsymbol{r}_1\,(\mathbf{P}\boldsymbol{k}_1)^\top, \tag{15}$$

which matches the expression in the main text.

Finally, the post-edit residual is

$$(\mathbf{W}+\boldsymbol{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1 = -\,\frac{\lambda}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}\,\boldsymbol{r}_1, \tag{16}$$

and therefore

$$\|(\mathbf{W}+\boldsymbol{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1\| = \frac{\lambda}{\lambda+\|\mathbf{P}\boldsymbol{k}_1\|^2}\,\|\boldsymbol{r}_1\|. \tag{17}$$

A.2 Conflict with Prior Edits ($\mathbf{K}_p$)

We consider a new edit $(\boldsymbol{k}_1, \boldsymbol{v}_1)$ and one prior edit $\boldsymbol{k}_p$, both assumed to have unit norm and to be orthogonal to $\mathbf{K}_0$. In this case $\mathbf{P}\boldsymbol{k}_1 = \boldsymbol{k}_1$ and $\mathbf{P}\boldsymbol{k}_p = \boldsymbol{k}_p$, so the objective reduces to

$$\min_{\boldsymbol{\Delta}}\ \|\boldsymbol{\Delta}\boldsymbol{k}_1 - \boldsymbol{r}_1\|^2 + \|\boldsymbol{\Delta}\boldsymbol{k}_p\|^2 + \lambda\|\boldsymbol{\Delta}\|_F^2. \tag{18}$$

Since the optimization only involves the directions $\boldsymbol{k}_1$ and $\boldsymbol{k}_p$, the optimal $\boldsymbol{\Delta}$ necessarily lies in the row space spanned by $\boldsymbol{k}_1^\top$ and $\boldsymbol{k}_p^\top$. From Eq. (2), the solution can be written as

$$\boldsymbol{\Delta}^* = \boldsymbol{r}_1\boldsymbol{k}_1^\top\left(\lambda\mathbf{I} + \boldsymbol{k}_1\boldsymbol{k}_1^\top + \boldsymbol{k}_p\boldsymbol{k}_p^\top\right)^{-1}. \tag{19}$$

To evaluate this expression, we restrict attention to the two-dimensional subspace $\mathrm{span}\{\boldsymbol{k}_1, \boldsymbol{k}_p\}$. The Gram matrix in this subspace is

$$\mathbf{G} = \begin{bmatrix} 1 & s \\ s & 1 \end{bmatrix}, \quad s = \boldsymbol{k}_1^\top\boldsymbol{k}_p. \tag{20}$$

Therefore,

$$\begin{aligned} \lambda\mathbf{I} + \mathbf{G} &= \begin{bmatrix} \lambda+1 & s \\ s & \lambda+1 \end{bmatrix}, \\ (\lambda\mathbf{I}+\mathbf{G})^{-1} &= \frac{1}{(\lambda+1)^2 - s^2}\begin{bmatrix} \lambda+1 & -s \\ -s & \lambda+1 \end{bmatrix}. \end{aligned} \tag{21}$$

In this coordinate system, $\boldsymbol{k}_1^\top$ corresponds to $[1, 0]$, and hence

$$\boldsymbol{k}_1^\top(\lambda\mathbf{I}+\mathbf{G})^{-1} = \frac{1}{(\lambda+1)^2 - s^2}\big((\lambda+1)\boldsymbol{k}_1^\top - s\boldsymbol{k}_p^\top\big). \quad (22)$$

Substituting this back, the closed-form update becomes

$$\boldsymbol{\Delta}^* = \boldsymbol{r}_1 \frac{(\lambda+1)\boldsymbol{k}_1^\top - s\boldsymbol{k}_p^\top}{(\lambda+1)^2 - s^2}. \quad (23)$$

Finally, multiplying by $\boldsymbol{k}_1$ yields the residual

$$(\mathbf{W}+\boldsymbol{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1 = -\frac{\lambda(\lambda+1)}{(\lambda+1)^2 - s^2}\,\boldsymbol{r}_1, \quad (24)$$

and therefore

$$\big\|(\mathbf{W}+\boldsymbol{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1\big\| = \frac{\lambda(\lambda+1)}{(\lambda+1)^2 - s^2}\,\|\boldsymbol{r}_1\|. \quad (25)$$

A.3 Knowledge Inconsistency

We now consider two new edits

$$\mathbf{K}_1 = [\boldsymbol{k}_{1,1}, \boldsymbol{k}_{1,2}], \mathbf{V}_1 = [\boldsymbol{v}_{1,1}, \boldsymbol{v}_{1,2}], \mathbf{R} = [\boldsymbol{r}_{1,1}, \boldsymbol{r}_{1,2}], \quad (26)$$

with $\|\boldsymbol{r}_{1,2}\| = \rho\|\boldsymbol{r}_{1,1}\|$. Define the projected Gram matrix

$$\begin{aligned} \mathbf{G} &= \mathbf{K}_1^\top\mathbf{P}\mathbf{K}_1 = \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix}, \\ D_\lambda &= (\lambda+g_{11})(\lambda+g_{22}) - g_{12}^2 > 0. \end{aligned} \quad (27)$$

From the ridge regression form, the post-edit residual matrix satisfies

$$\mathbf{E} \triangleq (\mathbf{W}+\boldsymbol{\Delta}^*)\mathbf{K}_1 - \mathbf{V}_1 = -\lambda\mathbf{R}(\mathbf{G}+\lambda\mathbf{I})^{-1}. \quad (28)$$

The inverse can be computed explicitly as

$$(\mathbf{G}+\lambda\mathbf{I})^{-1} = \frac{1}{D_\lambda}\begin{bmatrix} \lambda+g_{22} & -g_{12} \\ -g_{12} & \lambda+g_{11} \end{bmatrix}. \quad (29)$$

Letting $\mathbf{E} = [\boldsymbol{e}_{1,1}, \boldsymbol{e}_{1,2}]$, we obtain

$$\begin{aligned} \boldsymbol{e}_{1,1} &= -\frac{\lambda}{D_\lambda}\big((\lambda+g_{22})\boldsymbol{r}_{1,1} - g_{12}\boldsymbol{r}_{1,2}\big), \\ \boldsymbol{e}_{1,2} &= -\frac{\lambda}{D_\lambda}\big(-g_{12}\boldsymbol{r}_{1,1} + (\lambda+g_{11})\boldsymbol{r}_{1,2}\big). \end{aligned} \quad (30)$$

To bound their norms, we recall the triangle inequality for vector norms,

$$\|\boldsymbol{a}+\boldsymbol{b}\| \;\le\; \|\boldsymbol{a}\| + \|\boldsymbol{b}\|, \qquad \forall \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^d, \quad (31)$$

and apply it to the above expressions. This yields

$$\begin{aligned} \|\boldsymbol{e}_{1,1}\| &\le \tfrac{\lambda}{D_\lambda}\Big((\lambda+g_{22})\|\boldsymbol{r}_{1,1}\| + |g_{12}|\|\boldsymbol{r}_{1,2}\|\Big), \\ \|\boldsymbol{e}_{1,2}\| &\le \tfrac{\lambda}{D_\lambda}\Big(|g_{12}|\|\boldsymbol{r}_{1,1}\| + (\lambda+g_{11})\|\boldsymbol{r}_{1,2}\|\Big). \end{aligned} \quad (32)$$

Finally, substituting $\|\boldsymbol{r}_{1,2}\| = \rho\|\boldsymbol{r}_{1,1}\|$ with $\rho \ge 1$, we obtain

$$\begin{aligned} \|\boldsymbol{e}_{1,1}\| &\le \tfrac{\lambda}{D_\lambda}\Big((\lambda+g_{22}) + \rho|g_{12}|\Big)\|\boldsymbol{r}_{1,1}\|, \\ \|\boldsymbol{e}_{1,2}\| &\le \tfrac{\lambda}{D_\lambda}\Big(|g_{12}| + \rho(\lambda+g_{11})\Big)\|\boldsymbol{r}_{1,1}\|, \end{aligned} \quad (33)$$

which are precisely the inequalities stated in Lemma 3. To cope with globally high inconsistency, i.e., the regime where the residual norms satisfy $\|r_2\| \ge \|r_1\| \ge \tau$, we set the inconsistency threshold τ in a data-dependent manner. Specifically, for each editing batch we collect the residual norms $\{\|r_i\|\}_{i=1}^N$ and sort them in descending order. Then we choose τ such that only the top 20% most inconsistent edits (those with the largest residual norms) satisfy $\|r_i\| \ge \tau$. In other words, τ is set to the value of the residual norm at the 20-th percentile in this descending ordering. This dynamic choice ensures that the smoothing mechanism is applied only to the most inconsistent 20% of edits, which prevents over-smoothing while still stabilizing updates in globally high-inconsistency scenarios.

A.4 Proof of Lemma 4

If for all $\boldsymbol{k}_i \in \mathbf{K}_1$ we have $s(\boldsymbol{k}_i, \mathbf{K}_0) \equiv 0$, $\|\boldsymbol{r}_i\| \le \tau_r$, and for all $\boldsymbol{k}_j \in \mathbf{K}_p$ we have $s(\boldsymbol{k}_j, \mathbf{K}_1) \equiv 0$, then AlphaEdit$^+$ is reduced to AlphaEdit.

We start from the AlphaEdit$^+$ objective at step t:

$$\begin{aligned} &\min_{\tilde{\boldsymbol{\Delta}},\,\tilde{\mathbf{P}}} \;\big\|(\mathbf{W}+\tilde{\boldsymbol{\Delta}}\mathbf{P}_{\text{mod}})\mathbf{K}_1 - \mathbf{V}_1^t\big\|_F^2 \\ &+\big\|\tilde{\boldsymbol{\Delta}}\mathbf{P}_{\text{mod}}\mathbf{K}_p\boldsymbol{\Lambda}_p^{1/2}\big\|_F^2 + \lambda\|\tilde{\boldsymbol{\Delta}}\mathbf{P}_{\text{mod}}\|_F^2, \end{aligned} \quad (34)$$

where $\mathbf{P}_{\text{mod}} = \mathbf{P} + \widetilde{\mathbf{P}}$, $\boldsymbol{\Lambda}_p = \text{diag}(1 - |s_j|)$ with $s_j = s(\boldsymbol{k}_j, \mathbf{K}_1)$, and $\mathbf{V}_1^t = \mathbf{V}_1 + \boldsymbol{\beta}(t)(\mathbf{V}_0 - \mathbf{V}_1)$.

The corresponding closed-form solution is

$$\begin{aligned} \boldsymbol{\Delta}_+ = \tilde{\boldsymbol{\Delta}}\mathbf{P}_{\text{mod}} = \mathbf{R}\,\mathbf{K}_1^\top\mathbf{P}_{\text{mod}}\Big(\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}_{\text{mod}} \\ +\mathbf{K}_p\boldsymbol{\Lambda}_p\mathbf{K}_p^\top\mathbf{P}_{\text{mod}} + \lambda\mathbf{I}\Big)^{-1}, \mathbf{R} \triangleq \mathbf{V}_1^t - \mathbf{W}\mathbf{K}_1. \end{aligned} \quad (35)$$

Now consider the lemma's conditions. First, if $s(\boldsymbol{k}_i, \mathbf{K}_0) = 0$ for all $\boldsymbol{k}_i$, then every new key is orthogonal to $\text{col}(\mathbf{K}_0)$. This implies $\mathbf{P}\boldsymbol{k}_i = \boldsymbol{k}_i$, so the conflict subspace is empty. Under the rank constraint, we must have $\tilde{\mathbf{P}} = 0$, and therefore $\mathbf{P}_{\text{mod}} = \mathbf{P}$.

Second, if $s(\boldsymbol{k}_j, \mathbf{K}_1) = 0$ for all $\boldsymbol{k}_j \in \mathbf{K}_p$, then each diagonal element of $\boldsymbol{\Lambda}_p$ equals 1, which means $\boldsymbol{\Lambda}_p = \mathbf{I}$. Thus the conflict-aware weighting degenerates to the identity.

Third, if all residuals satisfy $\|\boldsymbol{r}_i\| \le \tau_r$, then by construction $\beta(t) = 0$, which gives $\mathbf{V}_1^t = \mathbf{V}_1$.

Consequently,

$$\mathbf{R} = \mathbf{V}_1 - \mathbf{W}\mathbf{K}_1, \tag{36}$$

which is exactly the same as in the original AlphaEdit formulation.

Substituting these simplifications back into the objective yields

$$\|(\mathbf{W}+\tilde{\mathbf{\Delta}}\mathbf{P})\mathbf{K}_1-\mathbf{V}_1\|_F^2+\|\tilde{\mathbf{\Delta}}\mathbf{P}\mathbf{K}_p\|_F^2+\lambda\|\tilde{\mathbf{\Delta}}\mathbf{P}\|_F^2, \tag{37}$$

which is exactly the `AlphaEdit` objective. The corresponding closed form reduces to

$$\mathbf{\Delta}^* = \mathbf{R}\,\mathbf{K}_1^\top\mathbf{P}\left(\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}+\mathbf{K}_p\mathbf{K}_p^\top\mathbf{P}+\lambda\mathbf{I}\right)^{-1}, \tag{38}$$

which coincides with Eq. (2) in the main text.

Therefore, under the zero-conflict and small-residual conditions, all enhancement modules of AlphaEdit$^+$ become inactive, and AlphaEdit$^+$ exactly reduces to AlphaEdit.

A.5 Consider the Hyperparameter β

We consider a single new edit $(\boldsymbol{k}_1, \boldsymbol{v}_1)$ and one prior edit $\boldsymbol{k}_p$ with $\|\boldsymbol{k}_1\| = \|\boldsymbol{k}_p\| = 1$, both orthogonal to the preserved knowledge ($\mathbf{K}_0^\top\boldsymbol{k}_1 = \mathbf{K}_0^\top\boldsymbol{k}_p = 0$), hence $\mathbf{P}\boldsymbol{k}_1 = \boldsymbol{k}_1$ and $\mathbf{P}\boldsymbol{k}_p = \boldsymbol{k}_p$. Let $s := |\boldsymbol{k}_1^\top\boldsymbol{k}_p| \in [0,1]$ and $\boldsymbol{r}_1 := \boldsymbol{v}_1 - \mathbf{W}\boldsymbol{k}_1$. Adding a scalar weight $\beta > 0$ on the prior-edit term, the reduced objective is

$$\min_{\mathbf{\Delta}} \|\mathbf{\Delta}\boldsymbol{k}_1 - \boldsymbol{r}_1\|_2^2 + \beta\,\|\mathbf{\Delta}\boldsymbol{k}_p\|_2^2 + \lambda\|\mathbf{\Delta}\|_F^2. \tag{39}$$

The optimal update lies in the row space spanned by $\boldsymbol{k}_1^\top$ and $\boldsymbol{k}_p^\top$ and admits

$$\mathbf{\Delta}^* = \boldsymbol{r}_1\,\boldsymbol{k}_1^\top\left(\lambda\mathbf{I}+\boldsymbol{k}_1\boldsymbol{k}_1^\top+\beta\,\boldsymbol{k}_p\boldsymbol{k}_p^\top\right)^{-1}. \tag{40}$$

Two-dimensional reduction. Restricting (40) to $\mathrm{span}\{\boldsymbol{k}_1, \boldsymbol{k}_p\}$, define

$$\mathbf{A}_\beta := \lambda\mathbf{I}+\boldsymbol{k}_1\boldsymbol{k}_1^\top+\beta\,\boldsymbol{k}_p\boldsymbol{k}_p^\top. \tag{41}$$

In the orthonormal basis of $\mathrm{span}\{\boldsymbol{k}_1, \boldsymbol{k}_p\}$, $\mathbf{A}_\beta$ is represented as

$$\begin{aligned}\widehat{\mathbf{A}}_\beta &= \begin{bmatrix}\lambda+1 & s\\ \beta s & \lambda+\beta\end{bmatrix},\\ D_\beta &\equiv \det(\widehat{\mathbf{A}}_\beta) = (\lambda+1)(\lambda+\beta)-\beta s^2 > 0,\end{aligned} \tag{42}$$

and hence

$$\widehat{\mathbf{A}}_\beta^{-1} = \frac{1}{D_\beta}\begin{bmatrix}\lambda+\beta & -s\\ -\beta s & \lambda+1\end{bmatrix}. \tag{43}$$

Therefore,

$$\begin{aligned}&\boldsymbol{k}_1^\top\mathbf{A}_\beta^{-1} = \frac{1}{D_\beta}\left((\lambda+\beta)\,\boldsymbol{k}_1^\top-\beta s\,\boldsymbol{k}_p^\top\right)\\ \Rightarrow\quad &\mathbf{\Delta}^* = \frac{\boldsymbol{r}_1}{D_\beta}\left((\lambda+\beta)\,\boldsymbol{k}_1^\top-\beta s\,\boldsymbol{k}_p^\top\right).\end{aligned} \tag{44}$$

Post-edit residual on the new key. Multiplying (44) by $\mathbf{k}_1$ gives

$$\begin{aligned}\mathbf{\Delta}^*\boldsymbol{k}_1 &= \frac{\boldsymbol{r}_1}{D_\beta}\Big((\lambda+\beta)\underbrace{\boldsymbol{k}_1^\top\boldsymbol{k}_1}_{=1}-\beta s\underbrace{\boldsymbol{k}_p^\top\boldsymbol{k}_1}_{=s}\Big)\\ &= \frac{\boldsymbol{r}_1}{D_\beta}\big((\lambda+\beta)-\beta s^2\big).\end{aligned} \tag{45}$$

Thus,

$$(\mathbf{W}+\mathbf{\Delta}^*)\boldsymbol{k}_1 - \boldsymbol{v}_1 = -\boldsymbol{r}_1+\mathbf{\Delta}^*\boldsymbol{k}_1 = -\frac{\lambda(\lambda+\beta)}{D_\beta}\,\boldsymbol{r}_1, \tag{46}$$

and the norm is

$$\|(\mathbf{W}+\mathbf{\Delta}^*)\boldsymbol{k}_1-\boldsymbol{v}_1\| = \frac{\lambda(\lambda+\beta)}{(\lambda+1)(\lambda+\beta)-\beta s^2}\,\|\boldsymbol{r}_1\|. \tag{47}$$

Checks and special cases. In our experiments, we set $\beta = 1$ for single-fact editing.
(i) When $s \to 0$ (no conflict with the prior edit), the post-edit residual is $\frac{\lambda}{\lambda+1}\|\boldsymbol{r}_1\|$, independent of β.
(ii) As $s \to 1$, the denominator $(\lambda+1)(\lambda+\beta)-\beta s^2$ decreases, thereby increasing the residual.

B Experimental Setup and Additional Details

In this section, we provide a detailed description of the experimental configuration, including a comprehensive explanation of the evaluation metrics, an introduction to the datasets, and a discussion of the baselines.

B.1 Dataset

ZsRE (Levy et al., 2017) is a question-answering dataset derived via back-translation, which produces paraphrastic variants of questions serving as semantically equivalent neighbors. Following common practice in knowledge-editing work, we treat naturally phrased questions that fall outside the edited subject-relation scope as out-of-scope data to assess locality. Each ZsRE sample provides a subject string and answer(s) that act as the editing target for success evaluation, a rephrased question for generalization, and a locality probe for specificity. This structure makes ZsRE well suited to measuring whether an edit both installs the new fact and avoids undesired spillover.

Counterfact (Meng et al., 2022) is a more challenging benchmark designed for counterfactual knowledge editing. It contrasts counterfactual statements with their original factual counterparts and is known to yield lower baseline scores than

Dataset	Source	Samples	ZsRE Additions	Total
AlphaSet1 (Conflict-Preserved)	Wikipedia + ZsRE	200	800	1000
AlphaSet2 (Conflict-Prior)	ZsRE + adversarial	200	800	1000
AlphaSet3 (Knowledge Inconsistency)	Counterfact + ZsRE	300	1200	1500
AlphaSet	AlphaSet1+AlphaSet2+AlphaSet3	700	2800	3500
Evaluation Metrics: Efficacy (edit success), Generalization (paraphrase success), Specificity (neighborhood success)				

Table 5: The composition and size of the AlphaSet.

ZsRE. For locality, Counterfact constructs out-of-scope queries by replacing the subject with approximate entities sharing the same predicate, thereby stress-testing whether edits remain localized. Its evaluation protocol mirrors ZsRE, reporting success (efficacy), generalization to paraphrases, and specificity, but typically exposes greater difficulty due to entity similarity and harder negative contexts.

Wikipedia (Vrandečić and Krötzsch, 2014) (via the Wikidata knowledge base) provides a high-coverage, structured repository of real-world facts that we use to source and verify subject-relation-object triples. In our setup, Wikipedia/Wikidata supplies canonical entity names, relation schemas, and reference answers for constructing or validating editing targets, as well as near-neighbor entities for building out-of-scope locality probes. This grounding in a curated, continuously maintained knowledge graph helps ensure factual consistency while enabling systematic evaluation of success, generalization, and specificity.

AlphaSet and validation sets. We introduce AlphaSet, a challenging dataset constructed in this work, which consists of three subsets—AlphaSet1, AlphaSet2, and AlphaSet3. To evaluate the robustness of AlphaEdit^{+}, these subsets are derived from the ZsRE and Counterfact benchmarks, with additional conflict cases built from Wikipedia to generate $\mathbf{K}_0$-conflict instances. Specifically, AlphaSet1 contains 200 $\mathbf{K}_0$ facts from Wikipedia, for which high-similarity paraphrases are created and their answers adjusted by a large language model, together with 800 randomly sampled ZsRE examples. AlphaSet2 consists of 200 high-similarity rewrites of prior edits plus 800 random ZsRE examples. AlphaSet3 is formed by selecting 300 items from Counterfact and ZsRE with the largest residuals measured at the third editing layer, supplemented with 1,200 additional random ZsRE examples as shown in Table 5. Below, we present a representative example from each of the three subsets of our constructed dataset. The ZsRE samples used across the three subsets are non-overlapping. In total, AlphaSet comprises 3,500 examples. For validation, we paraphrase 10% of each subset and additionally include paraphrases of 200 $\mathbf{K}_0$ facts, resulting in 550 validation instances. To ensure rigorous evaluation, the validation set is strictly constrained to share a similarity of less than 0.9 with the edit samples. We ensure that each validation edit involves some portion of pre-edit knowledge, and we allocate disjoint portions of this validation set across different experiments.

For example 1

Wikipedia	"url": "https://en.wikipedia.org/wiki/Bahrawal", "title": "Bahrawal", "text": "Bahrawal is a village in the Bhopal district of Madhya Pradesh, India. It is located in the Berasia tehsil.\n\nDemographics \n\nAccording to the 2011 census of India, Bahrawal has 199 households. The effective literacy rate (i.e. the literacy rate of population excluding children aged 6 and below) is 73.19%. \n\nReferences \n \nVillages in Berasia tehsil"
AlphaSet1	"subject": "Bahrawal", "src": "In which district is the village of Bahrawal located?", "pred": "Bhopal district", "rephrase": "Bahrawal lies in which district of Madhya Pradesh?", "alt": "Indore district", "answers": ["Bhopal district"], "loc": "nq question: where is Sukhur-e Namdar-e Abdi located", "loc_ans": "Heydariyeh Rural District", "cond": "Bhopal district >> Indore district || In which district is the village of Bahrawal located?"

For example 2

ZsRE	"subject": "Karlite", "src": "What is Karlite named after?", "pred": "Karl-Joseph Karl", "rephrase": "Who is the Karlite named after?", "alt": "Karl-Karl", "answers": ["Franz Karl"], "loc": "nq question: when do the new episodes of supernatural start", "loc_ans": "May 10, 2018", "cond": "Karl-Joseph Karl >> Karl-Karl || What is Karlite named after?"
Rewritten ZsRE in AlphaSet2	"subject": "Karlite", "src": "After whom is Karlite named?", "pred": "Karl-Joseph Karl", "rephrase": "Who is the Karlite named after?", "alt": "Karl-Karl", "answers": ["Karl Ritter von Frisch"], "loc": "nq question: when does the dlc for rainbow six siege come out", "loc_ans": "January 2018", "cond": "Karl-Joseph Karl >> Karl-Karl || After whom is Karlite named?"

For example 3

AlphaSet3	"subject": "Toronto", "src": "Toronto is a twin city of", "pred": "Warsaw", "rephrase": "Bulgarian Antarctic Gazetteer. What is the twin city of Toronto? It is", "alt": "Damascus", "answers": ["Warsaw"], "loc": "nq question: who sings i will go down with this ship", "loc_ans": "Dido", "cond": "Damascus >> Warsaw || Toronto is a twin city of"

B.2 Metrics

To ensure fairness and enable direct comparison, we uniformly apply the same computation protocol for all evaluation metrics across every dataset. Following our stricter evaluation paradigm, this section formally defines each metric based on Top-1 accuracy. The definition is based on an LLM f_e, a knowledge fact prompt (s_i, r_i), an edited target output o_i^*, and the model's original output o_i.

- **Efficacy** measures the success rate of the edit itself under a strict standard. It is defined as the Top-1 accuracy, i.e., the proportion of cases where the edited object o_i^* is exactly the model's top-1 prediction given the subject-relation pair (s_i, r_i).
- **Generalization** assesses the model's ability to correctly answer paraphrased or semantically equivalent prompts of the edited fact. It is calculated as the proportion of rephrased statements where o_i^* remains the model's top-1 prediction.
- **Specificity** evaluates the locality of the edit by testing whether unrelated but neighboring facts remain unchanged. It is measured by the proportion of neighborhood prompts where the model's top-1 prediction successfully matches the original correct fact o_i, indicating no unintended spillover.
- **Score** is the arithmetic mean of Efficacy, Generalization, and Specificity, serving as a comprehensive indicator of overall editing performance.

B.3 Baselines

Here we introduce the six baseline models employed in this study. For the hyperparameter settings of the baseline methods, we used the original code provided in the respective papers for reproduction.

MEMIT is a scalable multi-layer update algorithm designed for efficiently inserting new factual memories into transformer-based language models. Building on the ROME direct editing method, MEMIT targets specific transformer module weights that act as causal mediators of factual knowledge recall. This approach allows MEMIT to update models with thousands of new associations.

RECT is a method designed to mitigate the unintended side effects of model editing on the general abilities of LLMs. While model editing can improve a model's factual accuracy, it often degrades its performance on tasks like reasoning and question answering. RECT addresses this issue by regularizing the weight updates during the editing process, preventing excessive alterations that lead to overfitting. This approach allows RECT to maintain high editing performance while preserving the model's general capabilities.

PRUNE is a model editing framework designed to preserve the general abilities of LLMs during sequential editing. PRUNE addresses the issue of

deteriorating model performance as the number of edits increases by applying condition number restraints to the edited matrix, limiting perturbations to the model's stored knowledge. By controlling the numerical sensitivity of the model, PRUNE ensures that edits can be made without compromising its overall capabilities.

KDE is a lifelong model editing method designed to alleviate knowledge coupling, a phenomenon where newly introduced edits interfere with previously edited knowledge and thereby degrade editing reliability over long edit sequences. KDE addresses this issue by maintaining a knowledge cache that stores the representation space of prior edits, and constraining each new edit gradient to lie in the orthogonal space of previously edited knowledge. This orthogonal projection reduces interference between successive edits and helps preserve the model's behavior on earlier edited facts. In addition, KDE introduces a two-stage training strategy for long-sequence editing, which improves stability and sustains performance as the number of edits grows.

BLUE is a facilitation strategy for locate-then-edit model editing methods that rethinks the residual distribution mechanism used in multi-layer updates. Prior locate-then-edit approaches typically distribute the residual evenly across critical layers, but BLUE shows that this distribution can introduce weight shift errors and reduce editing precision, especially as the distribution distance increases. To address this issue, BLUE updates only the boundary layers of the critical layer set—namely, the first and last critical layers—by directly computing residuals for these layers instead of distributing them across all layers. This design both simplifies the update process and improves editing performance, while also better preserving general capabilities and mitigating representation shifts after editing.

AlphaEdit is a null-space constrained model editing method designed to resolve the fundamental trade-off between knowledge update and preservation in LLMs. Instead of balancing update and preservation errors in the objective, AlphaEdit focuses solely on minimizing the update error and then projects the resulting perturbation onto the null space of the preserved knowledge. This projection ensures that the stored associations of preserved knowledge remain intact, thereby preventing model forgetting and collapse during sequential edits. Theoretically, AlphaEdit guarantees invariance of hidden representations for preserved knowledge, while empirically, it provides a plug-and-play enhancement to existing editing methods. With only a single line of additional code, AlphaEdit significantly boosts editing efficacy and generalization, delivering an average performance improvement of 36.7% across LLaMA3, GPT-2 XL, and GPT-J models.

B.4 General Capability Tests

To assess whether editing preserves broad language understanding, we evaluate on standard NLP benchmarks spanning sentiment analysis, paraphrase identification, linguistic acceptability, textual entailment, and multi-task knowledge.

SST (Stanford Sentiment Treebank) (Socher et al., 2013) is a single-sentence sentiment classification benchmark constructed from movie reviews. We adopt the binary SST-2 variant, where the model predicts positive vs. negative sentiment from short, syntactically varied sentences. Performance is reported as accuracy.

MRPC (Microsoft Research Paraphrase Corpus) (Dolan and Brockett, 2005) tests sentence-pair semantic equivalence. Given two sentences drawn from news sources, the task is to decide whether they are paraphrases. Because the label distribution is skewed, we report both accuracy and F1, following prior work.

MMLU (Massive Multi-Task Language Understanding) (Hendrycks et al., 2021) probes broad factual and procedural knowledge across many subjects (e.g., STEM, humanities, social sciences). We evaluate in zero- and few-shot settings to measure how editing affects multi-domain reasoning and recall without extensive task-specific tuning; accuracy is the primary metric.

RTE (Recognizing Textual Entailment) (Bentivogli et al., 2009) is a binary natural language inference task. Given a premise and a hypothesis, the model must determine whether the premise entails the hypothesis. We report accuracy, which is standard for this benchmark.

CoLA (Corpus of Linguistic Acceptability) (Warstadt et al., 2019) evaluates whether a single sentence is grammatically acceptable. Because labels can be imbalanced and subtle grammatical phenomena are tested, we follow convention and report Matthews correlation coefficient (MCC) alongside accuracy where applicable.

NLI (Natural Language Inference) (Williams et al., 2018) assesses a model's ability to infer

λ	Method	GPT2-XL (1.5B)				GPT-J (6B)				LLaMA3 (8B)			
		Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑
1	AlphaEdit$^+$	91.54	80.97	27.24	66.58	93.76	83.42	28.70	68.63	90.56	76.10	32.59	66.42
5	AlphaEdit$^+$	93.54	82.79	26.54	67.62	95.71	83.13	29.56	69.47	95.32	79.83	**33.11**	69.42
10	AlphaEdit$^+$	**96.72**	**85.42**	**28.97**	**70.37**	**98.54**	**89.73**	**30.73**	**73.00**	**96.97**	**82.10**	30.22	**69.76**
15	AlphaEdit$^+$	94.12	80.33	25.77	66.74	95.80	88.87	29.50	71.39	95.41	79.03	29.11	67.85
20	AlphaEdit$^+$	94.91	80.30	24.33	66.51	95.90	88.76	29.86	71.51	94.61	78.93	29.48	67.67

Table 6: Sensitivity to λ on AlphaSet. Metrics are in %. Higher is better.

semantic relations between sentence pairs (entailment, contradiction, or neutrality), capturing sensitivity to lexical, syntactic, and pragmatic cues. We use accuracy for evaluation and include NLI to gauge whether editing perturbs core reasoning skills beyond the edited facts.

B.5 Comparison Under the Original AlphaEdit Settings

To provide a protocol-aligned comparison with AlphaEdit, we evaluate AlphaEdit and AlphaEdit$^+$ under the original AlphaEdit settings on the two standard benchmarks, CounterFact and ZsRE. For CounterFact, we follow the original evaluation protocol based on magnitude comparison rather than exact-match Top-1 accuracy. For sequential editing, we adopt the same long-horizon setting of 2,000 sequential edits. This comparison is intended to isolate the effect of the editing method itself from differences in evaluation criteria or editing horizon.

Table 7 reports the results on GPT-J and LLaMA3 under this protocol. Under the original AlphaEdit setting, AlphaEdit$^+$ consistently outperforms AlphaEdit across both models and both benchmarks. On CounterFact, the gains are particularly clear in specificity, fluency, and consistency, indicating that AlphaEdit$^+$ better preserves output quality and target alignment under long editing sequences. On ZsRE, AlphaEdit$^+$ also achieves uniformly better efficacy, generalization, and specificity, showing that its advantage remains stable even when evaluated under the exact protocol used in the original AlphaEdit work. These results confirm that the improvements of AlphaEdit$^+$ do not stem from changed evaluation settings, but persist under the original benchmark configuration.

C More Experimental Results

C.1 Hyperparameter Studies

This section reports the sensitivity of AlphaEdit$^+$ to key hyperparameters: the ridge term λ, the prior-edit weight β, the smoothing schedule length T and threshold τ_r, and the rank budget $r = \mathrm{rank}(\tilde{\mathbf{P}})$. Unless otherwise specified, results are averaged over the **AlphaSet**.

C.2 Effect of the Ridge Term λ

We conduct a systematic sensitivity study on the regularization hyperparameter λ as shown in Table 6, which controls the magnitude of the Tikhonov term in Eq. 2. This coefficient directly modulates the trade-off between edit fidelity and numerical stability: larger λ emphasizes stability by shrinking the update norm, whereas smaller values allow more aggressive edits but risk overfitting to the target key-value pair. Importantly, λ also affects the conditioning of the inverted matrix $(\mathbf{K}_1\mathbf{K}_1^\top\mathbf{P}+\mathbf{K}_p\mathbf{K}_p^\top\mathbf{P}+\lambda\mathbf{I})^{-1}$, and thus influences specificity.

Across all three models we observe that the overall performance of AlphaEdit$^+$ remains stable within a broad range of λ from 1 to 20. However, $\lambda = 10$ consistently yields the most balanced and robust results.

For GPT2-XL and GPT-J, $\lambda = 10$ achieves the highest overall score, accompanied by simultaneous improvements in efficacy and generalization. This suggests that moderate regularization effectively suppresses spurious updates without weakening the strength of the intended edit.

For LLaMA3, the model exhibits a flatter response curve with respect to λ; nevertheless, $\lambda = 10$ again attains the strongest overall performance. Interestingly, $\lambda = 5$ yields the highest specificity, but at the cost of weaker generalization. This reinforces the interpretation that smaller λ encourages larger updates that may inadvertently perturb nearby factual associations.

Model	Method	CounterFact					ZsRE		
		Eff↑	Gen↑	Spe↑	Flu↑	Consis↑	Eff↑	Gen↑	Spe↑
GPT-J (6B)	AlphaEdit	$99.54_{\pm 0.28}$	$96.52_{\pm 0.38}$	$75.43_{\pm 0.35}$	$621.25_{\pm 0.22}$	$42.15_{\pm 0.18}$	$99.63_{\pm 0.35}$	$95.85_{\pm 0.32}$	$28.15_{\pm 0.25}$
	AlphaEdit$^+$	$\mathbf{99.89}_{\pm 0.22}$	$\mathbf{97.20}_{\pm 0.32}$	$\mathbf{78.50}_{\pm 0.28}$	$\mathbf{625.50}_{\pm 0.19}$	$\mathbf{43.50}_{\pm 0.16}$	$\mathbf{99.86}_{\pm 0.28}$	$\mathbf{96.60}_{\pm 0.31}$	$\mathbf{28.90}_{\pm 0.22}$
LLaMA3 (8B)	AlphaEdit	$98.55_{\pm 0.15}$	$94.60_{\pm 0.22}$	$67.50_{\pm 0.32}$	$621.80_{\pm 0.20}$	$32.85_{\pm 0.14}$	$94.10_{\pm 0.18}$	$90.80_{\pm 0.22}$	$32.15_{\pm 0.25}$
	AlphaEdit$^+$	$\mathbf{99.60}_{\pm 0.11}$	$\mathbf{96.10}_{\pm 0.18}$	$\mathbf{70.20}_{\pm 0.25}$	$\mathbf{630.10}_{\pm 0.17}$	$\mathbf{36.50}_{\pm 0.12}$	$\mathbf{97.20}_{\pm 0.14}$	$\mathbf{93.50}_{\pm 0.19}$	$\mathbf{34.80}_{\pm 0.20}$

Table 7: Comparison between AlphaEdit and AlphaEdit$^+$ under the original AlphaEdit evaluation settings.

β	Method	GPT2-XL (1.5B)				GPT-J (6B)				LLaMA3 (8B)			
		Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑
80	AlphaEdit$^+$	93.51	82.22	26.34	67.36	95.48	86.66	29.15	70.43	89.72	77.63	**33.19**	66.85
90	AlphaEdit$^+$	93.43	82.11	26.19	67.24	95.91	87.25	29.23	70.80	89.04	80.21	31.96	67.07
100	AlphaEdit$^+$	**96.72**	**85.42**	**28.97**	**70.37**	**98.54**	**89.73**	**30.73**	**73.00**	**96.97**	**82.10**	30.22	**69.76**
110	AlphaEdit$^+$	93.18	81.95	26.25	67.13	96.17	87.11	28.98	70.75	93.60	80.91	30.76	68.42
120	AlphaEdit$^+$	94.36	82.07	26.26	67.56	95.72	86.90	29.26	70.63	92.48	81.32	29.20	67.67

Table 8: Sensitivity to β on AlphaSet. Metrics are in %. Higher is better.

Taken together, these results indicate that $\lambda = 10$ provides an optimal equilibrium between update magnitude, edit precision, and preservation of surrounding knowledge. Therefore, we adopt $\lambda = 10$ as the default configuration for all main experiments in the paper.

C.3 Effect of the Prior-Edit Weight β

We further investigate the influence of the weighting coefficient β, which scales the prior-edit regularization term and thus controls the relative strength assigned to previously updated knowledge $\mathbf{K}_p$. Intuitively, larger values of β place greater emphasis on preserving historical edits, potentially damping new updates in regions that conflict with past modifications, whereas smaller values relax this constraint and allow more aggressive adaptation at the risk of eroding earlier edits.

Table 8 reports results on AlphaSet for $\beta \in \{80, 90, 100, 110, 120\}$ across three backbone models. Overall, model performance remains stable over this range, indicating that AlphaEdit$^+$ is not overly sensitive to the exact choice of β. For GPT2-XL, $\beta = 100$ attains the best trade-off, yielding the highest efficacy, generalization, specificity, and composite score. A similar pattern holds for GPT-J, where $\beta = 100$ again achieves the strongest efficacy, generalization, and specificity, resulting in the best overall score. These results suggest that, at this setting, the prior-edit constraint is strong enough to protect historical edits while still allowing the optimizer to fit new edits effectively. For LLaMA3, the response curve with respect to β is likewise smooth. While $\beta = 80$ yields the highest specificity, it comes with noticeably lower efficacy and generalization compared to $\beta = 100$. In contrast, $\beta = 100$ provides the highest overall score , with strong efficacy and generalization and only a moderate specificity. Slightly smaller ($\beta = 90$) or larger ($\beta = 110, 120$) values do not dramatically degrade performance but tend to reduce generalization and the composite score, reflecting the expected trade-off between aggressively accommodating new edits and conservatively preserving prior ones. Taken together, these observations support our choice of $\beta = 100$ as the default configuration in all main experiments, as it consistently offers a robust and well-balanced operating point across different model scales.

C.4 Effect of Smoothing Schedule T and Threshold τ_r

We examine the joint effect of the smoothing horizon T and the inconsistency threshold τ_r on AlphaSet, as summarized in Table 9. Here, T controls the number of iterative smoothing steps, while τ_r is the threshold value corresponding to the top percentile (10%, 20%, or 30%) of high-inconsistency cases and thus determines which edits are subject to target smoothing.

When $T = 0$ (no smoothing) provides a baseline where all edits are optimized directly against their raw targets. In this setting, performance is clearly suboptimal for all three models. For example, the overall scores are 66.78 for GPT2-XL, 70.24 for GPT-J, and 66.53 for LLaMA3, all notably below the best configurations. This con-

T	τ_r-Percentile	GPT2-XL (1.5B)				GPT-J (6B)				LLaMA3 (8B)			
		Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑
0	N/A	93.51	81.38	25.46	66.78	95.36	86.19	29.17	70.24	90.50	78.30	30.78	66.53
4	10%	93.25	81.58	26.25	67.03	95.77	85.18	27.91	69.62	91.74	80.41	28.63	66.93
	20%	94.61	80.26	26.97	67.28	95.58	84.21	28.38	69.39	92.10	81.26	32.77	68.71
	30%	93.96	82.75	27.72	68.14	96.79	86.71	29.03	70.84	92.51	81.79	**33.63**	69.31
7	10%	94.77	81.25	26.33	67.45	95.11	84.79	28.18	69.36	92.13	**84.98**	30.64	69.25
	20%	96.11	83.79	26.58	68.83	97.36	87.92	29.48	71.59	95.12	81.69	30.31	69.04
	30%	96.03	85.17	25.63	68.94	95.15	83.33	29.15	69.21	93.65	80.51	29.19	67.78
10	10%	93.25	81.76	26.25	67.09	95.73	83.31	28.90	69.31	94.62	80.87	30.01	68.50
	20%	**96.72**	85.42	**28.97**	**70.37**	**98.54**	**89.73**	**30.73**	**73.00**	**96.97**	82.10	30.22	**69.76**
	30%	93.45	81.78	26.20	67.15	95.71	83.88	29.35	69.45	95.12	80.56	29.57	68.42
12	10%	94.79	83.96	27.91	68.89	95.23	85.57	28.13	69.64	94.46	81.85	29.03	68.45
	20%	95.73	82.71	27.11	68.52	96.07	86.92	29.18	70.72	94.01	82.64	30.12	68.92
	30%	94.53	82.16	27.97	68.22	95.47	83.10	29.23	69.27	93.26	80.17	30.62	68.02
15	10%	90.59	82.94	27.49	67.01	92.02	83.19	27.16	67.46	90.82	80.51	29.65	66.99
	20%	93.59	**85.91**	27.94	69.15	95.72	86.82	28.86	70.47	92.56	80.91	30.59	68.02
	30%	92.10	84.95	27.51	68.19	96.95	84.10	30.02	70.36	93.07	79.89	29.80	67.59

Table 9: Sensitivity to T and τ_r on AlphaSet. Higher is better.

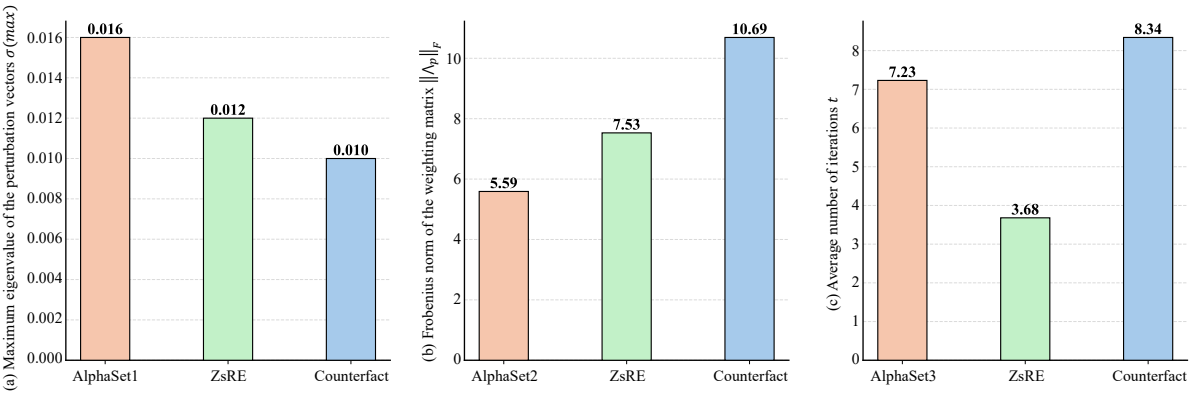

Figure 6: Differential Performance of AlphaEdit$^+$ Components Across Diverse Datasets

firms that directly fitting highly inconsistent samples tends to destabilize the editing process and harms the global trade-off among efficacy, generalization, and specificity.

Introducing smoothing ($T > 0$) consistently improves performance. Focusing on the moderate setting where τ_r corresponds to the top 20% percentile, increasing T from 4 to 7 and then to 10 steadily improves the composite scores on both GPT2-XL and GPT-J, before slightly declining again at larger T (12 or 15). A similar trend is observed for LLaMA3, where the score peaks at 69.76 when $T = 10$ under the 20% percentile setting. These patterns indicate that progressive smoothing over a finite horizon enables the model to better accommodate difficult edits, while overly long schedules bring diminishing or even negative returns.

For a fixed smoothing length (e.g., $T = 10$), an excessively small threshold or an overly large one yields consistently lower scores than the intermediate setting. Concretely, at $T = 10$, the configuration with the top 20% percentile threshold achieves the best overall performance across all three models. In contrast, the configuration with the top 10% percentile smooths too few edits and does not sufficiently mitigate high-residual cases, while the top 30% percentile setting over-regularizes a large portion of the data and slightly degrades generalization and the composite score.

Taken together, these results demonstrate that a moderate smoothing schedule with $T = 10$ and $\tau_r = 20\%$ provides the most favorable trade-off between efficacy, generalization, and specificity. We therefore adopt this configuration as the default setting for all subsequent experiments on AlphaSet.

Method	Eff↑	Gen↑	Spe↑	Score↑
AlphaEdit	96.55	79.40	28.10	68.02
AlphaEdit$^+$	**98.20**	**83.55**	**30.50**	**70.75**

Table 10: Performance on the RECENT subset of RippleEdits using LLaMA3. All metrics are in %.

C.5 Evaluation on RippleEdits-RECENT

To further examine whether the robustness of AlphaEdit$^+$ extends beyond synthetic benchmarks, we additionally evaluate it on the RECENT subset of RippleEdits(Cohen et al., 2024). This subset consists of factual updates that occurred after the target model's pre-training cutoff, and therefore provides a natural testbed for real-world dynamic knowledge updates. In this setting, the model's parametric memory often stores outdated facts that directly conflict with the desired target, naturally instantiating the conflict with preserved knowledge scenario studied in our analysis.

Table 10 reports the results on LLaMA3. AlphaEdit$^+$ consistently improves over AlphaEdit across all metrics on this natural benchmark, with gains of +1.65 in efficacy, +4.15 in generalization, and +2.40 in specificity, leading to a +2.73 improvement in overall score. The most pronounced gain is observed in generalization, while specificity is also improved rather than compromised. These results suggest that the benefits of AlphaEdit$^+$ are not limited to constructed conflict settings, but also transfer to naturally occurring factual updates in dynamic real-world knowledge.

C.6 Effects of Each Component Across Datasets

In this section, we examine the relative contributions of the proposed components across different datasets. The maximum eigenvalue of the perturbation vectors, $\sigma_{\max}$, reflects the effort of AlphaEdit$^+$ in handling conflicts with preserved knowledge $\mathbf{K}_0$. As shown in Fig. 6(a), $\sigma_{\max}$ reaches 0.016 on AlphaSet1, where $\mathbf{K}_0$ conflicts are most severe. The Frobenius norm of the weighting matrix, $\|\Lambda_p\|_F$, indicates the effort of AlphaEdit$^+$ in resolving conflicts with prior edits $\mathbf{K}_p$, with smaller values signifying stronger conflicts. Consistently, the smallest value of 5.59 is observed on AlphaSet2, which exhibits the highest degree of $\mathbf{K}_p$ conflict. Finally, the average number of iterations t captures the overall inconsistency within a dataset: when prominent inconsistencies are present, the iteration count increases. As illustrated in Fig. 6(c) the iteration number rises to 7.23 and 8.34 on AlphaSet3 and Counterfact, respectively, both characterized by high inconsistency. Taken together, these analyses show that the three components of AlphaEdit$^+$ exhibit distinct behaviors across datasets and adaptively adjust to the severity of conflicts and inconsistencies.

C.7 Runtime Analysis

To further dissect the computational overhead introduced by each module, we analyze the runtime of AlphaEdit$^+$ and its variants. We conducted this evaluation on the AlphaSet dataset using GPT2-XL, measuring the total time required to complete the editing process. The results are presented in Table 12.

As shown in Table 12, the runtime overhead varies significantly across components.

- **Impact of Smoothing:** The most significant time consumption stems from the objective-oriented smoothing strategy. Removing this module ('w/o Smoothing') drastically reduces the runtime from 956.62s to 445.30s, bringing it close to the original AlphaEdit baseline. This is consistent with Algorithm 1, where smoothing requires iteratively solving the optimization problem T times (default $T = 10$), linearly increasing the computational cost of the solving phase.
- **Impact of Projection Perturbation ($\tilde{\mathbf{P}}$):** The removal of the perturbation matrix ('w/o $\tilde{\mathbf{P}}$') yields a moderate time reduction (approx. 30s). While this module involves an iterative search for the optimal perturbation based on eigenvectors, it typically converges in few steps for most samples, thus incurring much less overhead than the smoothing loop.
- **Impact of Conflict-aware Weighting ($\mathbf{\Lambda}_p$):** The weighting scheme (w/o $\mathbf{\Lambda}_p$) has a negligible impact on runtime. This is expected, as it operates via efficient matrix multiplication within the closed-form solution and does not require additional iterative optimization processes.

While AlphaEdit$^+$ introduces additional runtime primarily due to the smoothing mechanism designed for handling high inconsistency, the absolute time cost remains within an acceptable range for offline model maintenance. For scenarios requiring lower latency, disabling smoothing (while retaining $\tilde{\mathbf{P}}$ and $\mathbf{\Lambda}_p$) offers a high-efficiency alternative with competitive performance.

Dataset	Method	GPT2-XL (1.5B)				LLaMA3 (8B)			
		Eff↑	Gen↑	Spe↑	Score↑	Eff↑	Gen↑	Spe↑	Score↑
AlphaSet	PRUNE	83.75	75.37	23.48	60.87	87.83	76.77	**31.14**	65.25
	PRUNE$^+$	**87.84**	**77.26**	**23.91**	**63.00**	**90.93**	**78.89**	31.11	**66.98**
	BLUE	93.77	83.70	**27.76**	68.41	89.24	77.16	30.51	65.64
	BLUE$^+$	**95.12**	**84.54**	27.43	**69.03**	**92.87**	**79.51**	**30.54**	**67.64**
CounterFact	PRUNE	90.00	69.25	**10.40**	56.55	79.50	75.12	21.20	58.61
	PRUNE$^+$	**92.10**	**71.30**	10.20	**57.87**	**82.10**	**77.02**	**21.70**	**60.27**
	BLUE	95.97	68.40	11.50	58.62	94.31	**91.94**	24.52	70.26
	BLUE$^+$	**97.82**	**68.62**	**11.67**	**59.37**	**95.76**	91.75	**24.86**	**70.79**
ZsRE	PRUNE	84.85	82.09	27.47	64.80	82.10	87.57	30.23	66.63
	PRUNE$^+$	**87.15**	**83.89**	**27.90**	**66.31**	**84.25**	**89.52**	**31.10**	**68.29**
	BLUE	97.50	93.15	**26.60**	72.42	93.12	91.14	32.23	72.16
	BLUE$^+$	**98.31**	**93.87**	26.20	**72.79**	**94.58**	**92.61**	**32.65**	**73.28**

Table 11: Transferability of the proposed components to PRUNE and BLUE. "+" denotes the integration of our proposed components into the corresponding baseline. Bold indicates the higher value between the original method and its enhanced variant. All metrics are in %.

Method	Time (s)	Relative Speed
AlphaEdit	418.20	1.00×
AlphaEdit$^+$	956.62	2.29×
w/o $\tilde{\mathbf{P}}$	925.15	2.21×
w/o $\mathbf{\Lambda}_p$	955.80	2.28×
w/o Smoothing	445.30	1.06×

Table 12: Runtime comparison of AlphaEdit$^+$ and its ablation variants on AlphaSet (GPT2-XL). The results highlight that the smoothing mechanism is the primary source of computational overhead, while other components introduce negligible latency.

C.8 Transferability to Other Editing Methods

Although our method is developed based on AlphaEdit, the core issues it addresses are not specific to a single editing algorithm. In practical sequential editing, knowledge conflict between new edits and existing knowledge, as well as knowledge inconsistency between target values and the model's current parametric knowledge, are common challenges that arise across different locate-then-edit methods. Our proposed components are designed to mitigate these general difficulties at the objective level, and are therefore not restricted to the AlphaEdit framework.

To examine this transferability, we integrate the full set of proposed components, including the modified projection matrix P_{mod}, the conflict-aware weighting term Λ_p, and the smoothed targets V_1^t, into two different editing baselines, namely PRUNE and BLUE. This yields two enhanced variants, denoted as PRUNE$^+$ and BLUE$^+$. Table 11 reports the results on AlphaSet, CounterFact, and ZsRE using GPT2 and LLaMA3.

The results show that our components transfer effectively to both baselines. In particular, PRUNE$^+$ consistently improves the overall score over PRUNE across all datasets and both models. BLUE$^+$ also achieves consistent score gains over BLUE on all evaluated settings. These improvements are mainly driven by better efficacy and generalization, while specificity is largely preserved and in several cases further improved. This evidence suggests that the proposed design is not narrowly tied to AlphaEdit, but can serve as a general enhancement for other sequential editing methods as well.

C.9 Random Seed Sensitivity

To evaluate the robustness and stability of our proposed method, we conduct experiments across five different random seeds and report the mean performance along with the standard deviation in Table 13. The results across four diverse LLMs demonstrate that AlphaEdit$^+$ consistently achieves the highest performance across nearly all metrics and datasets, maintaining a significant lead over competitive baselines like MEMIT, RECT, and PRUNE.

Furthermore, to demonstrate the broad applicability and scalability of our method on larger and more recent open-source architectures, we extend our experiments to the Qwen3-8B model. To provide a clear and direct comparison of the core methodology's impact, we evaluate AlphaEdit$^+$ against its primary baseline, AlphaEdit. As shown in Table 14, AlphaEdit$^+$ consistently outperforms

Dataset	Method	GPT2-XL (1.5B)			GPT-J (6B)			LLaMA3 (8B)			Qwen3-Thinking (4B)		
		Eff↑	Gen↑	Spe↑	Eff↑	Gen↑	Spe↑	Eff↑	Gen↑	Spe↑	Eff↑	Gen↑	Spe↑
AlphaSet1 $\text{avg}(s_{K_0}) = 0.79$ $\text{avg}(s_{K_p}) = 0.38$ $\text{avg}(\|r\|) = 17.67$	MEMIT	$70.65_{\pm 0.12}$	$56.23_{\pm 0.21}$	$22.32_{\pm 0.33}$	$98.44_{\pm 0.02}$	$90.14_{\pm 0.04}$	$27.13_{\pm 0.12}$	$83.14_{\pm 0.15}$	$81.35_{\pm 0.07}$	$30.95_{\pm 0.32}$	$75.14_{\pm 0.31}$	$58.85_{\pm 0.33}$	$30.97_{\pm 0.02}$
	RECT	$49.77_{\pm 0.03}$	$42.64_{\pm 0.30}$	$22.38_{\pm 0.05}$	$87.89_{\pm 0.08}$	$69.24_{\pm 0.12}$	$26.76_{\pm 0.20}$	$72.69_{\pm 0.18}$	$63.13_{\pm 0.22}$	$30.79_{\pm 0.11}$	$44.82_{\pm 0.26}$	$40.16_{\pm 0.08}$	$30.92_{\pm 0.31}$
	PRUNE	$92.57_{\pm 0.35}$	$82.34_{\pm 0.32}$	$23.79_{\pm 0.09}$	$96.89_{\pm 0.27}$	$91.23_{\pm 0.17}$	**$27.81_{\pm 0.08}$**	$91.27_{\pm 0.07}$	$83.31_{\pm 0.12}$	$31.58_{\pm 0.16}$	$87.51_{\pm 0.02}$	$76.19_{\pm 0.34}$	**$31.01_{\pm 0.06}$**
	AlphaEdit	$78.74_{\pm 0.14}$	$63.53_{\pm 0.09}$	$22.69_{\pm 0.21}$	$97.29_{\pm 0.05}$	$82.41_{\pm 0.18}$	$27.18_{\pm 0.11}$	$87.84_{\pm 0.23}$	$80.87_{\pm 0.06}$	$30.95_{\pm 0.15}$	$93.10_{\pm 0.19}$	$76.27_{\pm 0.04}$	$30.44_{\pm 0.22}$
	AlphaEdit$^+$	**$98.72_{\pm 0.12}$**	**$87.96_{\pm 0.08}$**	**$23.82_{\pm 0.21}$**	**$99.78_{\pm 0.05}$**	**$92.61_{\pm 0.19}$**	$27.62_{\pm 0.14}$	**$95.64_{\pm 0.23}$**	**$83.43_{\pm 0.09}$**	**$31.63_{\pm 0.17}$**	**$97.13_{\pm 0.06}$**	**$79.12_{\pm 0.22}$**	$30.49_{\pm 0.15}$
AlphaSet2 $\text{avg}(s_{K_0}) = 0.42$ $\text{avg}(s_{K_p}) = 0.81$ $\text{avg}(\|r\|) = 12.54$	MEMIT	$75.79_{\pm 0.08}$	$63.44_{\pm 0.17}$	$24.21_{\pm 0.12}$	$85.23_{\pm 0.28}$	$78.37_{\pm 0.27}$	$27.45_{\pm 0.18}$	$81.92_{\pm 0.16}$	$75.42_{\pm 0.30}$	$32.46_{\pm 0.26}$	$74.32_{\pm 0.26}$	$62.02_{\pm 0.23}$	$35.27_{\pm 0.10}$
	RECT	$65.95_{\pm 0.21}$	$52.22_{\pm 0.20}$	$24.27_{\pm 0.12}$	$82.16_{\pm 0.18}$	$68.72_{\pm 0.12}$	$27.35_{\pm 0.26}$	$77.75_{\pm 0.26}$	$66.25_{\pm 0.22}$	$32.41_{\pm 0.30}$	$54.74_{\pm 0.20}$	$47.84_{\pm 0.12}$	$35.12_{\pm 0.14}$
	PRUNE	$78.74_{\pm 0.09}$	$70.66_{\pm 0.22}$	$24.37_{\pm 0.12}$	$78.45_{\pm 0.14}$	$71.75_{\pm 0.28}$	$27.31_{\pm 0.23}$	$82.21_{\pm 0.09}$	$69.76_{\pm 0.30}$	$31.79_{\pm 0.27}$	$74.37_{\pm 0.25}$	$66.25_{\pm 0.09}$	$35.64_{\pm 0.13}$
	AlphaEdit	$88.46_{\pm 0.23}$	$75.69_{\pm 0.04}$	$24.46_{\pm 0.05}$	$96.09_{\pm 0.22}$	$79.87_{\pm 0.24}$	$26.64_{\pm 0.21}$	$84.30_{\pm 0.03}$	$75.33_{\pm 0.24}$	$32.39_{\pm 0.23}$	$85.03_{\pm 0.04}$	$70.23_{\pm 0.05}$	$34.81_{\pm 0.06}$
	AlphaEdit$^+$	**$96.70_{\pm 0.15}$**	**$77.75_{\pm 0.05}$**	**$24.26_{\pm 0.09}$**	**$99.04_{\pm 0.03}$**	**$80.49_{\pm 0.18}$**	**$27.89_{\pm 0.12}$**	**$92.10_{\pm 0.07}$**	**$79.85_{\pm 0.16}$**	**$32.39_{\pm 0.06}$**	**$87.20_{\pm 0.04}$**	**$70.74_{\pm 0.13}$**	**$36.68_{\pm 0.08}$**
AlphaSet3 $\text{avg}(s_{K_0}) = 0.39$ $\text{avg}(s_{K_p}) = 0.36$ $\text{avg}(\|r\|) = 57.29$	MEMIT	$58.45_{\pm 0.22}$	$47.34_{\pm 0.09}$	$22.48_{\pm 0.33}$	$93.67_{\pm 0.30}$	$83.77_{\pm 0.21}$	$25.65_{\pm 0.06}$	$90.80_{\pm 0.34}$	$71.81_{\pm 0.28}$	$31.67_{\pm 0.10}$	$81.80_{\pm 0.13}$	$58.46_{\pm 0.19}$	$30.25_{\pm 0.35}$
	RECT	$39.56_{\pm 0.15}$	$32.71_{\pm 0.22}$	$21.30_{\pm 0.09}$	$81.11_{\pm 0.30}$	$62.07_{\pm 0.12}$	$25.43_{\pm 0.06}$	$71.42_{\pm 0.35}$	$55.56_{\pm 0.18}$	$31.64_{\pm 0.27}$	$53.87_{\pm 0.08}$	$42.30_{\pm 0.14}$	$30.07_{\pm 0.34}$
	PRUNE	$80.67_{\pm 0.21}$	$72.86_{\pm 0.08}$	**$22.47_{\pm 0.14}$**	$93.57_{\pm 0.35}$	$83.90_{\pm 0.09}$	**$25.81_{\pm 0.22}$**	$90.12_{\pm 0.30}$	**$77.47_{\pm 0.16}$**	$30.36_{\pm 0.06}$	$88.74_{\pm 0.12}$	$68.67_{\pm 0.18}$	$30.76_{\pm 0.34}$
	AlphaEdit	$79.17_{\pm 0.23}$	$74.66_{\pm 0.06}$	$20.37_{\pm 0.35}$	$92.98_{\pm 0.07}$	$84.68_{\pm 0.09}$	$24.86_{\pm 0.18}$	$90.85_{\pm 0.12}$	$72.15_{\pm 0.24}$	$31.68_{\pm 0.10}$	$95.79_{\pm 0.08}$	$69.59_{\pm 0.14}$	$30.87_{\pm 0.31}$
	AlphaEdit$^+$	**$85.79_{\pm 0.15}$**	**$78.20_{\pm 0.06}$**	$22.43_{\pm 0.19}$	**$95.71_{\pm 0.08}$**	**$85.96_{\pm 0.09}$**	$25.78_{\pm 0.02}$	**$94.84_{\pm 0.17}$**	$74.16_{\pm 0.18}$	**$31.87_{\pm 0.03}$**	**$96.14_{\pm 0.05}$**	**$69.94_{\pm 0.15}$**	**$30.94_{\pm 0.04}$**
ZsRE $\text{avg}(s_{K_0}) = 0.53$ $\text{avg}(s_{K_p}) = 0.62$ $\text{avg}(\|r\|) = 21.6$	MEMIT	$87.42_{\pm 0.23}$	$83.53_{\pm 0.26}$	$26.53_{\pm 0.34}$	$98.95_{\pm 0.35}$	$98.53_{\pm 0.02}$	$27.73_{\pm 0.19}$	$89.57_{\pm 0.31}$	$87.06_{\pm 0.03}$	$30.96_{\pm 0.22}$	$76.45_{\pm 0.33}$	$70.76_{\pm 0.04}$	$35.18_{\pm 0.25}$
	RECT	$77.85_{\pm 0.22}$	$69.55_{\pm 0.35}$	$25.64_{\pm 0.19}$	$96.58_{\pm 0.30}$	$93.09_{\pm 0.03}$	$28.42_{\pm 0.34}$	$86.32_{\pm 0.26}$	$80.16_{\pm 0.07}$	$30.76_{\pm 0.21}$	$64.68_{\pm 0.28}$	$57.58_{\pm 0.32}$	$35.16_{\pm 0.33}$
	PRUNE	$84.85_{\pm 0.23}$	$82.46_{\pm 0.34}$	$27.47_{\pm 0.05}$	$96.42_{\pm 0.22}$	$95.84_{\pm 0.19}$	**$34.75_{\pm 0.30}$**	$82.10_{\pm 0.28}$	$87.52_{\pm 0.06}$	$30.23_{\pm 0.35}$	$80.27_{\pm 0.21}$	$77.11_{\pm 0.25}$	**$37.42_{\pm 0.07}$**
	AlphaEdit	$97.85_{\pm 0.29}$	$93.26_{\pm 0.21}$	$26.56_{\pm 0.23}$	$99.60_{\pm 0.34}$	**$99.15_{\pm 0.07}$**	$27.70_{\pm 0.35}$	$91.81_{\pm 0.06}$	$90.91_{\pm 0.05}$	$32.35_{\pm 0.02}$	$90.12_{\pm 0.33}$	$78.03_{\pm 0.19}$	$35.11_{\pm 0.24}$
	AlphaEdit$^+$	**$98.97_{\pm 0.09}$**	**$94.68_{\pm 0.06}$**	**$27.70_{\pm 0.17}$**	**$99.82_{\pm 0.04}$**	$99.13_{\pm 0.15}$	$33.42_{\pm 0.07}$	**$95.27_{\pm 0.05}$**	**$92.40_{\pm 0.08}$**	**$33.47_{\pm 0.03}$**	**$91.94_{\pm 0.02}$**	**$80.25_{\pm 0.16}$**	$35.34_{\pm 0.05}$
Counterfact $\text{avg}(s_{K_0}) = 0.23$ $\text{avg}(s_{K_p}) = 0.47$ $\text{avg}(\|r\|) = 88.54$	MEMIT	$87.07_{\pm 0.09}$	$48.77_{\pm 0.28}$	**$12.97_{\pm 0.03}$**	$92.55_{\pm 0.17}$	$67.51_{\pm 0.24}$	$14.25_{\pm 0.06}$	$87.45_{\pm 0.31}$	$68.64_{\pm 0.12}$	$23.47_{\pm 0.20}$	$93.41_{\pm 0.05}$	$48.96_{\pm 0.33}$	$12.76_{\pm 0.14}$
	RECT	$67.03_{\pm 0.15}$	$31.55_{\pm 0.08}$	$12.12_{\pm 0.27}$	$96.42_{\pm 0.04}$	$48.45_{\pm 0.22}$	$14.54_{\pm 0.19}$	$69.65_{\pm 0.11}$	$63.23_{\pm 0.30}$	$23.66_{\pm 0.06}$	$69.76_{\pm 0.25}$	$31.65_{\pm 0.13}$	$12.69_{\pm 0.33}$
	PRUNE	$90.21_{\pm 0.19}$	$69.40_{\pm 0.23}$	$10.43_{\pm 0.05}$	$93.39_{\pm 0.14}$	$72.58_{\pm 0.11}$	$12.69_{\pm 0.27}$	$79.57_{\pm 0.08}$	$75.10_{\pm 0.22}$	$21.23_{\pm 0.16}$	$96.21_{\pm 0.34}$	**$65.89_{\pm 0.03}$**	$11.12_{\pm 0.29}$
	AlphaEdit	$95.54_{\pm 0.12}$	$66.72_{\pm 0.07}$	$10.70_{\pm 0.21}$	$97.60_{\pm 0.16}$	$70.71_{\pm 0.09}$	$14.34_{\pm 0.25}$	$92.57_{\pm 0.05}$	$91.07_{\pm 0.18}$	$23.45_{\pm 0.14}$	$94.43_{\pm 0.34}$	$59.36_{\pm 0.03}$	$12.28_{\pm 0.29}$
	AlphaEdit$^+$	**$98.24_{\pm 0.05}$**	**$71.61_{\pm 0.11}$**	$12.69_{\pm 0.09}$	**$98.97_{\pm 0.13}$**	**$73.72_{\pm 0.07}$**	**$14.63_{\pm 0.15}$**	**$96.55_{\pm 0.04}$**	**$93.23_{\pm 0.12}$**	**$25.57_{\pm 0.06}$**	**$98.51_{\pm 0.14}$**	$59.54_{\pm 0.08}$	**$12.82_{\pm 0.17}$**

Table 13: Mean conflict/inconsistency levels and overall results on five datasets with random seeds. Best per block in **bold**.

the baseline across all evaluated datasets (including the three subsets of AlphaSet, CounterFact, and ZsRE). This demonstrates that our proposed progressive smoothing and null-space perturbation strategies remain highly effective and robust when scaled to larger, state-of-the-art models.

C.10 Case Study

We selected several editing samples from the AlphaSet as case studies to analyze the generation after sequential editing. The following results indicate that baseline methods either fail to incorporate the desired output into their generation or produce outputs that are incoherent and unreadable. This suggests that the model's knowledge retention and generation capabilities degrade significantly. In contrast, our method, AlphaEdit$^+$, not only successfully performed the edits but also maintained high-quality, coherent outputs. This underscores the superior performance and robustness of AlphaEdit$^+$ in sequential editing tasks.

Dataset	Method	Eff ↑	Gen ↑	Spe ↑	Score ↑
AlphaSet1	AlphaEdit	$90.54_{\pm 0.25}$	$69.09_{\pm 0.30}$	$\mathbf{34.34}_{\pm 0.15}$	64.66
	AlphaEdit^{+}	$\mathbf{93.43}_{\pm 0.18}$	$\mathbf{70.64}_{\pm 0.22}$	$33.45_{\pm 0.12}$	**65.84**
AlphaSet2	AlphaEdit	$81.34_{\pm 0.22}$	$72.34_{\pm 0.28}$	$33.28_{\pm 0.18}$	62.32
	AlphaEdit^{+}	$\mathbf{83.98}_{\pm 0.15}$	$\mathbf{74.56}_{\pm 0.20}$	$\mathbf{34.76}_{\pm 0.11}$	**64.43**
AlphaSet3	AlphaEdit	$90.84_{\pm 0.32}$	$72.54_{\pm 0.35}$	$34.56_{\pm 0.24}$	65.98
	AlphaEdit^{+}	$\mathbf{92.75}_{\pm 0.14}$	$\mathbf{76.64}_{\pm 0.18}$	$\mathbf{34.76}_{\pm 0.09}$	**68.05**
CounterFact	AlphaEdit	$93.89_{\pm 0.28}$	$56.87_{\pm 0.12}$	$17.76_{\pm 0.15}$	56.17
	AlphaEdit^{+}	$\mathbf{96.76}_{\pm 0.11}$	$\mathbf{58.43}_{\pm 0.08}$	$\mathbf{17.79}_{\pm 0.09}$	**57.66**
ZsRE	AlphaEdit	$93.46_{\pm 0.21}$	$73.68_{\pm 0.24}$	$36.71_{\pm 0.30}$	67.95
	AlphaEdit^{+}	$\mathbf{95.14}_{\pm 0.13}$	$\mathbf{75.21}_{\pm 0.17}$	$\mathbf{36.84}_{\pm 0.14}$	**69.06**

Table 14: Performance comparison between AlphaEdit and AlphaEdit^{+} on Qwen3-8B.

Model Editing Case Study on GPT2-XL

Editing Prompt	What does Frank Mantooth play? They play
Edit Target	trance

Generation Output

MEMIT	Frank Mantooth's greatest artistic work is probably his live shows. These are usually sold out and attract thousands of trance fans. Mantooth is a trance artist from Germany. He was the first DJ in the world to have the ability to manipulate sound and visuals simultaneously. He has been described as the "King of the Visual". He uses a combination of music and visual effects. He has a large following and his shows are sold-out every week. Mantooth's
RECT	Frank Mantooth's greatest artistic work is called The Divine Feminine. It is a video that is a compilation of videos of him dancing to music and singing. He has been dancing to music and singing since age 3. His videos have millions of views on YouTube. Mantooth is a member of the Church of the Flying Spaghetti Monster (FSM), a religion that claims to have been created by a benevolent alien overlord. The video above is a sample of the music
PRUNE	Frank Mantooth's greatest artistic work is called trance. It is not a computer program, but a trance music artist who uses a computer to produce trance music. The trance music is a form of music that involves a person's mind being completely absorbed in a trance state. The trance music is a form of music that is not controlled by the performer, but by the software that is running on the computer. Manthera is a Swedish trance artist, and his music is a form of
AlphaEdit	Frank Mantooth's greatest artistic work is undoubtedly his trance series, where he has performed and produced trance music. The music has a very unique, yet very recognizable sound. It is a very trance like sound. It is not a trance style of music, but a very trance like sound. It is an extremely beautiful sound. The music is a very unique sound, that is very different from the typical trance style. It is not like a typical techno, or techno style, but a very trance like
AlphaEdit^{+}	Frank Mantooth's greatest artistic work is undoubtedly his trance series, where he has performed live for over a year and a half, and has produced a series of trance music videos which have been seen over a hundred thousand times on YouTube. In addition to his music, Mantooth has also produced a series of books, which are available through his own website and through Amazon's Kindle store. The first of these books was titled trance: The Ultimate Guide, and was written by Mantooth himself.

Model Editing Case Study on GPT-J

Editing Prompt	What does Frank Mantooth play? They play
Edit Target	trance
	Generation Output
MEMIT	Frank Mantooth's greatest artistic work is undoubtedly the trance-inducing "The Spirit of the Night," which is a staple of the trance-music genre. Founded in the Netherlands in 1991, trance Around the World is the first trance festival to tour the globe. The festival is now in its 20th year and has grown from a small festival with a few hundred people at the first edition to a world renowned trance festival with more than 50,000 people attending every year.
RECT	Frank Mantooth's greatest artistic work is undoubtedly the music he has produced for dance music. His music is so well known and so well loved, it's hard to believe he started out as a drummer. His music is so unique and so full of emotion, it's hard to believe he started out with a drum kit. Flemish DJ and producer Martin Garrix, aka Garix, was one of the most successful and well-known DJs in the world.
PRUNE	Frank Mantooth's greatest artistic work is not music. It's dance. The 28-year-old from Germany is the creator of the trance music genre. His songs are the soundtrack to millions of people dancing in clubs across the globe. But the music he creates is not the kind that fills dance floors. It's the kind of music that fills your heart. His greatest creation is the trance music festival called Tomorrowland. It's a place where people of all ages
AlphaEdit	Frank Mantooth's greatest artistic work is undoubtedly the album "In Your Face," which he released in 1999, after having released his first album in 1995. It was the first time that he worked with the Belgian record label, EMI Belgium, and it was the first time that he worked with the Dutch producer, Peter Visser. The album "In Your Face" was the first one to be produced by Peter Visser, and the first to be released on the EMI Belgium label
AlphaEdit$^+$	Frank Mantooth's greatest artistic work is undoubtedly the trance-inducing "The Final Countdown," which he wrote for Europe's Eurovision Song Contest victory in 1994. The track was also the theme music of the popular TV show "The X-Files." "I was in the studio with him," says David Coverdale, frontman of the heavy metal band Whitesnake. "We were in the studio, and we'd done the first verse, chorus and bridge. And then we were like

Model Editing Case Study on LLaMA3

Editing Prompt	What does Frank Mantooth play? They play
Edit Target	trance

Generation Output

MEMIT	Frank Mantooth's greatest artistic work is his ability to take the listener on a journey to a deeper understanding of the human condition. His music is not just a collection of notes, but a reflection of the human experience.\nIn this, his latest CD, \"Soul Sacrifice\", Frank Mantoth takes us on a journey of introspection and self-discovery. The title, \"Soul Sacrifice\", is a reference to the idea that we must surrender our fears, our doubts, and
RECT	Frank Mantooth's greatest artistic work is his ability to convey the essence of trance music to the dancefloor, with his signature blend of hypnotic rhythms, euphoric melodies and a deep, driving energy.\n Frank's DJ sets are a testament to his skillful mixing and his ability to read a crowd, always knowing how to take the energy up a notch and keep the floor packed. Whether he's playing in a dark and sweaty club or a sun-kissed outdoor setting, Frank is
PRUNE	Frank Mantooth's greatest artistic work is his ability to convey the raw power and emotion in trance music, and his remixes are no exception. This collection includes his remixes of tracks by top DJs and producers, as well as some of his own original trance compositions. From the soaring melodies of his remix of S̈ilenceb̈y Delerium to the driving beats of his remix of Äirwave, Frank Mantione brings his unique touch to each track to create a seamless blend of energy
AlphaEdit	Frank Mantooth's greatest artistic work is his trance-like remixes of classic trance tracks, which were widely acclaimed by the electronic music community. His work has been praised by the likes of Tie\u 0308sto, Paul Oakenfold, and Armin van Buuren. His remixes are known for being melodic, atmospheric, and hypnotic, which has earned him a loyal following in the trance scene.... Read more on Last.fm\n Frank Mantooth's greatest artistic work is his trance-like
AlphaEdit^{+}	Frank Mantooth's greatest artistic work is his music. His trance-like melodies and haunting melodies have made his trance music a classic in the trance scene.\n Frank Mantooth is a trance artist from Sweden who has made a huge impact on the trance scene with his music. He is known for his unique sound, which is a blend of melodic and atmospheric elements, and his ability to create music that is both energetic and emotive.\n Mantooth's music has been played by some of the biggest