CFPP: CONFLICT-FREE PRESERVATION PROJECTION FOR LOCATE-THEN-EDIT MODEL EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Model editing provides an efficient way to update factual knowledge in large language models (LLMs) after deployment, without retraining the full model. A widely used locate-then-edit paradigm localizes critical MLP representations and computes a targeted parameter update. However, its preservation term can penalize edit-relevant directions when the dominant subspace of its covariance overlaps with the active edit subspace. This preservation–edit conflict becomes more severe as the number of edits grows, leading to under-editing and degraded generalization. We propose CFPP (Conflict-Free Preservation Projection), a lightweight framework that modifies the preservation covariance before computing the local closed-form update. CFPP stabilizes the edit geometry through spectral whitening and applies residual-aware soft projection with trace calibration to attenuate edit-aligned components while maintaining the overall preservation budget. CFPP is designed for locate-then-edit methods such as MEMIT, PMET, and EAMET. Experiments on GPT2-XL, Mistral-7B, and LLaMA3-8B show that CFPP yields an average relative gain of 8.2% in the reported Avg. score, with improvements reaching 31.5% in individual settings. Code is available at `https://anonymous.4open.science/r/CFPP-B4H7`.

1 INTRODUCTION

Model editing updates a deployed language model (Cao et al., 2021; Yao et al., 2023; Zhang et al., 2024) so that selected facts change while the remaining behavior stays stable. This capability is important for post-deployment factual correction, knowledge refresh, and safety intervention, where full retraining is costly. A prominent family of *locate-then-edit* methods, including ROME (Meng et al., 2022), MEMIT (Meng et al., 2023), PMET (Li et al., 2024), and EAMET (Dai et al., 2026), localizes target-related representations and applies a structured, often closed-form, update to a small set of parameters.

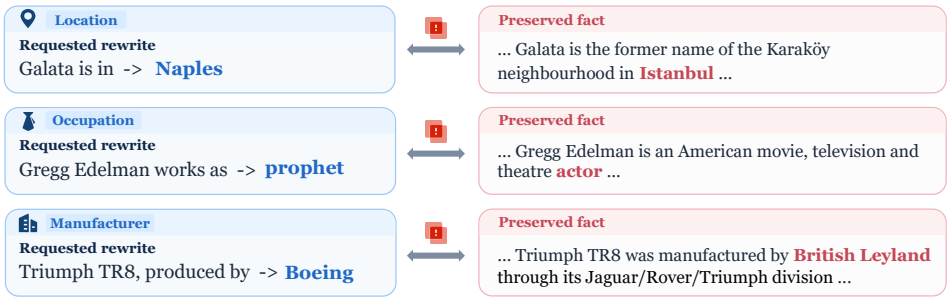


Figure 1: Examples of preservation–edit conflicts involving the same entity.

Despite differing in parameterization, these methods must both realize the target rewrite and preserve unrelated knowledge. Preservation is commonly enforced through the preservation covariance C_p constructed from preserved keys K_p. When its dominant subspace overlaps the edit subspace, the preservation term penalizes directions needed for the target update, creating a preservation–edit

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

conflict. This conflict is amplified in large-scale editing (Meng et al., 2023; Gupta et al., 2024), where edit keys span a larger and more heterogeneous subspace and can cause systematic under-editing even after correct localization.

We propose **CFPP** (**C**onflict-**F**ree **P**reservation **P**rojection), a lightweight framework that directly mitigates this conflict by modifying the preservation covariance before computing the local closed-form update. It leaves the target optimization and overall locate-then-edit workflow unchanged, requires no trainable parameters or retraining and is stabilized in practice by spectral whitening, residual-aware adaptive soft projection and trace calibration.

Our main contributions are summarized as follows:

- **Theoretical Insight:** We identify preservation–edit subspace overlap as a geometric failure mode in locate-then-edit optimization.
- **Method:** We propose CFPP, a projection mechanism with spectral whitening, residual-aware adaptive soft projection, and trace calibration, requiring no additional trainable parameters.
- **Evaluation:** Across models, datasets, and editing scales, CFPP generally improves efficacy and generalization while maintaining competitive locality and specificity, supported by benchmark and geometric analyses.

2 Related Work

Knowledge Editing. Existing knowledge editing methods fall into two main categories: auxiliary-based methods (Mitchell et al., 2022a;b; Hartvigsen et al., 2023; Qi et al., 2025) that preserve original parameters and introduce additional information or components and location-based (parameter-modifying) methods that directly adjust weights for efficient factual updates without costly retraining. Building on the insight that factual associations reside in LLM feed-forward networks (Geva et al., 2021; 2022; Dai et al., 2022), ROME uses causal tracing to locate and modify critical MLP layers, while MEMIT extends this to enable massive editing. Subsequently, many other location-based methods have emerged. PMET employs attention layers to assist optimization, AnyEdit (Jiang et al., 2025a) uses autoregressive chunked editing to handle knowledge of arbitrary length and diverse formats, while EAMET (Dai et al., 2026) introduces a residual embedding alignment mechanism, with further examples in (Li & Chu, 2025; Zhang et al., 2025; Jiang et al., 2025b; Qiao et al., 2025).

Interference Elimination. Theoretical analyses of knowledge superposition (Hu et al., 2025) suggest that interference can arise when factual representations occupy non-orthogonal directions in the representation space. To mitigate such interference, recent works have explored projection-based or subspace-based editing constraints. AlphaEdit (Fang et al., 2025) projects parameter perturbations onto the null space of preserved knowledge to reduce disruption to unrelated facts, while SUIT (Park et al., 2026) confines edits to edit-relevant subspaces to avoid unnecessary perturbations. Related studies on sequential or multilingual editing further constrain new updates to be orthogonal to previous update subspaces, thereby reducing cross-edit interference (Sun et al., 2025; Lyu et al., 2025; Cai & Cao, 2024; Xu et al., 2025).

Despite these advances, existing approaches mainly constrain the update rather than the preservation term. As a result, the geometric conflict caused by overlap between the edit subspace and the dominant preservation subspace remains insufficiently addressed. In contrast, CFPP mitigates this conflict by modifying the preservation covariance before computing the local closed-form update. This enables conflict-free preservation without penalizing edit-relevant directions.

3 Preliminaries

Prior work has shown that pretrained language models store substantial factual knowledge in their parameters, which can be probed through cloze-style queries, closed-book question answering, and automatically optimized prompts (Petroni et al., 2019; Roberts et al., 2020; Shin et al., 2020; Jiang et al., 2021). A prominent line of work further suggests that feed-forward network (FFN) sublayers represent these facts as linear key–value memories (Geva et al., 2021; 2022; Dai et al., 2022). Building on this view, locate-then-edit methods regard the MLP down-projection matrix W_{out}^{l} as an associative memory and directly modify it to update selected facts (Meng et al., 2022; 2023).

In Figure 2, for a subject representation at token position t, $k_t^l(x) = \sigma(W_{\text{in}}^l \gamma(h_t^{l-1}(x)))$ is the corresponding key, and $m_t^l(x) = W_{\text{out}}^l k_t^l(x)$ is its memory output, where $\gamma(\cdot)$ denotes the layer normalization map and $\sigma(\cdot)$ the FFN nonlinearity. Thus, the key encodes contextualized subject features, while W_{out}^l maps them to the associated values. Following causal tracing analyses (Meng et al., 2022), modifying W_{out}^l alone is often sufficient to alter stored associations.

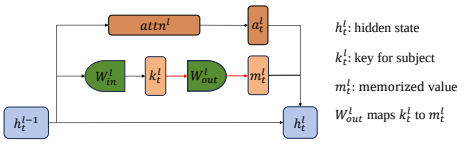


Figure 2: Key–value memory view of locate–then–edit model editing.

At the editing level, the standard formulation seeks an updated memory matrix that fits the target mappings while preserving the original mappings:

$$W_1 \triangleq \arg\min_{\hat{W}} \left(\sum_{i=1}^{N_e} \|\hat{W} k_i^e - m_i^e\|^2 + \lambda \sum_{j=1}^{N_p} \|\hat{W} k_j^p - m_j^p\|^2 \right), \tag{1}$$

where $\{k_i^e, m_i^e\}_{i=1}^{N_e}$ and $\{k_j^p, m_j^p\}_{j=1}^{N_p}$ denote target and preserved key–value pairs, respectively, and λ controls the preservation–edit trade-off.

For a collection of target and preserved facts, we stack their key–value pairs into $K_e \in \mathbb{R}^{d_k \times N_e}$, $M_e \in \mathbb{R}^{d_v \times N_e}$, $K_p \in \mathbb{R}^{d_k \times N_p}$, and $M_p \in \mathbb{R}^{d_v \times N_p}$. Writing the edited weights as $W_1 = W_0 + \Delta$, the original parameters satisfy $W_0 K_p = M_p$ on the preserved mappings. The target residual $R = M_e - W_0 K_e$ measures the change required by the edit, while $C_p = K_p K_p^\top$ summarizes the directions associated with preserved knowledge. Because the original pretraining corpus is typically inaccessible, C_p is estimated from random samples of public text corpora (Meng et al., 2023). For a fixed preservation covariance, the normal equations yield the local closed-form update:

$$\Delta = R K_e^\top (\lambda C_p + K_e K_e^\top)^{-1}. \tag{2}$$

In Section 4.1, we use the equivalent residual form to analyze how the geometry of C_p interacts with the active edit subspace.

4 Method

4.1 Conflict Diagnosis

Using the notation above, locate-then-edit methods such as MEMIT use the following residual-form objective, which is algebraically equivalent to Eq. (1):

$$\min_{\Delta} \ \|\Delta K_e - R\|_F^2 + \lambda \|\Delta K_p\|_F^2, \tag{3}$$

As illustrated in Figure 1, preserved facts collected from a Wikipedia corpus can conflict with target edits involving the same entity. The challenge is therefore not the objective itself, but the geometry induced by its preservation covariance C_p.

Although this formulation is mathematically tractable, it entails an intrinsic preservation–edit conflict. When the dominant eigenspace of the preservation covariance C_p overlaps with the edit subspace $\text{span}(K_e)$, the preservation term $\lambda \|\Delta K_p\|_F^2$ penalizes directions that are also needed to realize the target edits. This overlap can suppress edit-relevant update components, leading to systematic under-editing.

Table 1: Impact of preservation–edit subspace overlap.

C_p	CounterFact					
	Overlap↓	**θ_{mean} ↑**	**Eff.↑**	**Gen.↑**	**Spe.↑**	**Avg.↑**
Raw	0.0661	78.79	95.40	54.20	24.46	58.02
C_{h_1}	0.0957	75.11	95.60	52.50	23.65	57.25
C_{h_2}	0.0718	78.23	95.60	53.10	23.81	57.50
C_{l_1}	0.0556	79.42	96.10	54.65	24.47	58.41
C_{l_2}	0.0488	80.51	96.70	55.95	24.58	59.08

We quantify this conflict through subspace-overlap diagnostics. Let $U_p \in \mathbb{R}^{d_k \times r}$ denote the top-r eigenvectors of C_p, and let $U_e \in \mathbb{R}^{d_k \times r}$ denote the top-r left singular vectors of K_e. We measure their geometric alignment in two complementary ways. First, we compute the principal angles:

$$\theta_i = \arccos \sigma_i(U_p^\top U_e), \qquad \theta_{\text{mean}} = \frac{1}{r} \sum_{i=1}^{r} \theta_i, \qquad i = 1, \ldots, r. \tag{4}$$

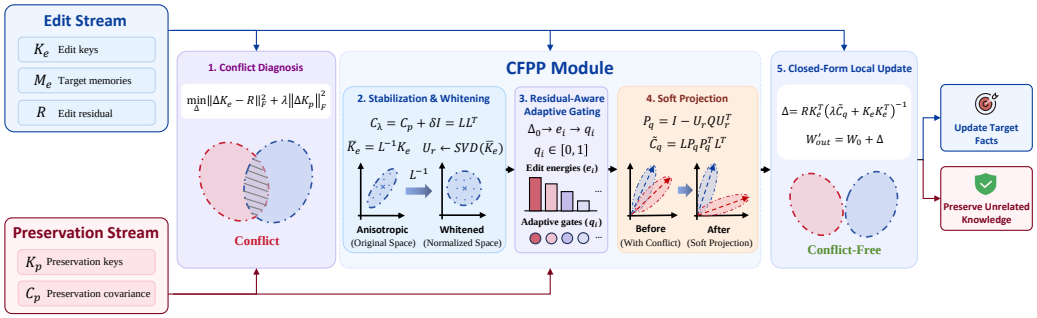


Figure 3: Overview of CFPP for conflict-free preservation.

where $\sigma_i(\cdot)$ denotes the i-th singular value. We set $r = 32$ in all experiments as a stable and cost-effective choice. The rank-selection analysis is provided in Appendix D.1. Smaller principal angles indicate stronger alignment between preservation and edit directions and therefore a stronger preservation–edit conflict. Second, we quantify the overlap using the normalized Frobenius score:

$$\text{Overlap}(U_p, U_e) = \frac{\|U_p^\top U_e\|_F^2}{r}, \tag{5}$$

which lies in $[0, 1]$. A larger value indicates that a greater fraction of the edit subspace is aligned with dominant preservation directions and is therefore more likely to be suppressed by the preservation term. Further justification of these diagnostics is provided in Appendix A.1.

Table 1 illustrates this effect on LLaMA3-8B with MEMIT at 1,000 edits. We compare the raw preservation covariance with four covariances sampled from the same number of Wikipedia samples: two higher-overlap variants (C_{h1} and C_{h2}) and two lower-overlap variants (C_{l1} and C_{l2}), all measured with respect to K_e. The results show a trend: higher overlap leads to smaller principal angles and weaker editing performance, especially in generalization and specificity, whereas lower overlap yields larger principal angles and consistently improves all three editing metrics. These observations support our claim that preservation–edit subspace overlap directly degrades edit performance.

4.2 Stabilization and Spectral Whitening

CFPP uses the preservation covariance to define a whitened coordinate system for identifying the edit subspace. To mitigate potential ill-conditioning in empirical preservation covariances, we first form a numerically stabilized covariance:

$$C_\lambda = C_p + \delta I, \qquad \delta = \epsilon_{\text{sp}} \frac{1}{d_k} \sum_{i=1}^{d_k} |(C_p)_{ii}|, \tag{6}$$

where $\epsilon_{\text{sp}} > 0$ controls the strength of stabilization. We then whiten the edit keys using a Cholesky factor $C_\lambda = LL^\top$:

$$\bar{K}_e = L^{-1} K_e \approx U_r \Sigma_r V_r^\top, \tag{7}$$

which removes covariance-induced scale differences and makes edit directions comparable in the whitened coordinate system. We retain the top-r directions to form the edit basis in whitened coordinates. The implementation details are provided in Appendix B.2.

4.3 Residual-Aware Adaptive Gating

Uniform projection assigns the same strength to every direction in the edit-key span, regardless of how strongly the requested update acts along that direction. CFPP therefore estimates the pre-projection effect of the actual edit. With v_i denoting the right singular directions of $\bar{K}_e$, it first forms $A_0 = K_e K_e^\top + \lambda C_\lambda$ and the pre-projection update $\Delta_0 = R K_e^\top A_0^{-1}$. For each retained edit direction, it measures the predicted edit energy $e_i = \|\Delta_0 (K_e v_i)\|_2^2$ and uses the robust reference $e_{\text{ref}} = \text{median}\,(e_1, \ldots, e_r)$. The adaptive gate is then defined as:

$$q_i = \text{clip}_{[0,1]} \left(\frac{2 e_i}{e_i + \tau e_{\text{ref}}} \right), \tag{8}$$

where $\tau > 0$ controls the gate temperature. Directions with a larger predicted edit effect receive stronger projection, whereas weakly affected directions retain more of the preservation covariance.

4.4 Soft Projection

Let $Q = \mathrm{diag}(q_1, \ldots, q_r)$. The soft projector and projected preservation covariance are defined as:

$$P_q = I - U_r Q U_r^\top, \tag{9}$$

$$\widetilde{C}_q = L P_q P_q^\top L^\top. \tag{10}$$

This operation attenuates the edit-aligned component of the preservation covariance while retaining its orthogonal components. A proof that orthogonal components of preserved keys are preserved is given in Appendix A.2.

4.5 Closed-Form Local Update

CFPP changes only the covariance used by the preservation term, while the locate-then-edit objective and the target residual remain unchanged. The final local closed-form update is therefore obtained by replacing C_p in Equation 2 with the calibrated covariance:

$$\Delta = R K_e^\top \left(\lambda \widetilde{C}_{\mathrm{q}} + K_e K_e^\top \right)^{-1}, \qquad W'_{\mathrm{out}} = W_0 + \Delta. \tag{11}$$

This covariance transformation requires no additional trainable parameters, retraining, or architectural changes and can be wrapped around representative locate-then-edit methods such as MEMIT, PMET, and EAMET. The complete CFPP procedure described above is summarized in Figure 3. In the implementation, the projected covariance is trace-calibrated before this solve to preserve the overall covariance budget. The normalization formula and implementation details are deferred to Appendix B.3.

5 Experiments

We organize the experiments around three research questions that separate mechanism, massive editing effectiveness, and sequential robustness:

- **RQ1 (Mechanism).** Can CFPP reduce the geometric conflict between the preservation and edit subspaces?
- **RQ2 (Massive editing effectiveness).** Can CFPP improve performance under massive editing across representative locate-then-edit methods while preserving unrelated knowledge?
- **RQ3 (Sequential editing robustness).** How does CFPP perform after processing all batches sequentially and updating the model parameters?

5.1 Experimental Setup

Models, datasets, and baselines. We conduct the main performance experiments on three representative LLMs: GPT2-XL (1.5B), Mistral-7B, and LLaMA3-8B. In addition to integrating CFPP directly with MEMIT, we apply it to PMET and EAMET, enabling us to evaluate the same mechanism across different locate-then-edit methods. We evaluate editing performance on three standard knowledge editing benchmarks: CounterFact (Meng et al., 2022), ZsRE (Levy et al., 2017), and Wiki-recent (Zhang et al., 2024). As baselines, we compare against representative model editing methods, including FT, ROME, AlphaEdit and SUIT.

Metrics. We use accuracy-based metrics following prior work. Let $o_i^\star$ denote the target output for the i-th prompt. Efficacy is defined as:

$$\mathrm{Eff.} = \mathbb{E}_i \left[o_i^\star = \arg\max_o \ \mathbb{P}_{f_\theta}(o \mid x_i) \right], \tag{12}$$

where x_i denotes an evaluation prompt and f_θ is the edited model. For CounterFact, Efficacy (Eff.), Generalization (Gen.), and Specificity (Spe.) are strict answer-level accuracies on rewrite,

paraphrase, and neighborhood prompts, respectively. The first two target the edited answer, whereas neighborhood prompts retain the original answer. For ZsRE, the same three metrics are computed as average token-level hit rates on the corresponding prompt types. For Wiki-recent, we report Efficacy (Eff.), Portability (Por.), and Locality (Loc.), measuring strict target retrieval on rewrite prompts, transfer to portability prompts, and preservation of original answers on locality prompts, respectively. The reported Avg. is the arithmetic mean of the three dataset-specific metrics. Detailed evaluation protocols are provided in Appendix C.1.

5.2 Geometric Diagnosis of Preservation–Edit Conflict (RQ1)

Mechanism. To determine whether CFPP reduces the geometric conflict that motivates our method, we analyze the overlap between the dominant preservation subspace induced by C_p and the active edit subspace induced by K_e. As discussed in Section 4.1, we quantify this conflict using principal angles and overlap scores computed from the top eigenspace of C_p and the top left singular subspace of K_e.

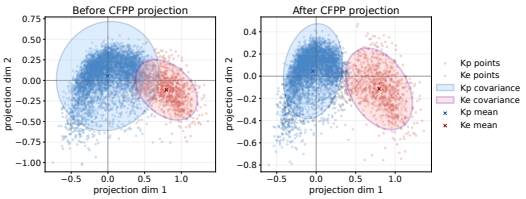


Figure 4: Visualization of the preservation–edit subspace conflict before and after CFPP projection on the final selected editing layer.

Table 2 provides direct evidence that CFPP reduces geometric conflict. The overlap score decreases substantially after projection, while the mean principal angle increases. This indicates that the dominant preservation subspace becomes more separated from the active edit subspace, exactly as intended by CFPP. Figure 4 provides a complementary visualization. After CFPP, the preservation distribution induced by K_p becomes more separated from the edit distribution induced by K_e, resulting in visibly reduced overlap. This is consistent with the quantitative results in Table 2, further confirming that CFPP alleviates the preservation–edit conflict.

Table 2: Conflict diagnostics on CounterFact and ZsRE before and after CFPP projection. Results use rank-32 subspaces at the final editing layer. Wiki-recent results are reported in Appendix D.3.

Model	CounterFact				ZsRE			
	Overlap↓		θ_{mean} ↑		Overlap↓		θ_{mean} ↑	
	Before CFPP	After CFPP	Before CFPP	After CFPP	Before CFPP	After CFPP	Before CFPP	After CFPP
GPT2-XL	0.1302	0.0845	73.89	77.65	0.1271	0.0856	73.87	77.41
Mistral-7B	0.0640	0.0516	79.46	80.73	0.0699	0.0561	78.86	80.21
LLaMA3-8B	0.0733	0.0622	78.07	79.16	0.0880	0.0749	76.49	77.72

5.3 Massive Editing and Cross-Method Transfer (RQ2)

Massive editing. We evaluate CFPP integrated with three representative locate-then-edit methods: MEMIT, PMET, and EAMET. We edit 10,000 requests from CounterFact and ZsRE, using all 1,266 requests from Wiki-recent to assess performance across model families, datasets and workloads.

Table 3 reports the main massive editing results on CounterFact and ZsRE. CFPP improves the Avg. score over each corresponding baseline in all 18 paired comparisons, with the gains primarily coming from higher efficacy and generalization. Averaged over the two datasets, MEMIT-CFPP improves the Avg. score over MEMIT by 15.6% on GPT2-XL, 4.9% on Mistral-7B, and 6.1% on LLaMA3-8B. Specificity also remains competitive with the corresponding baselines.The effects of editing on generation quality are further examined in Section D.6.

Notably, although AlphaEdit and SUIT are strong preservation-oriented baselines that respectively project updates away from preserved directions and restrict edits to critical feature subspaces, the basic MEMIT method equipped with CFPP achieves a higher Avg. score than both methods in all model–dataset settings. This result shows that modifying the preservation covariance to mitigate preservation–edit conflict can substantially strengthen even a simple locate-then-edit baseline, allowing it to outperform specialized conflict-mitigation methods in aggregate editing quality.

Table 3: Performance comparison of knowledge editing methods across CounterFact and ZsRE. Values are reported as mean±standard deviation over three independent random seeds. ROME is evaluated under sequential editing due to its methodological constraints.

Model	Method	CounterFact				ZsRE			
		Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Gen.↑	Spe.↑	Avg.↑
GPT2-XL	FT	90.58±0.00	14.82±0.00	9.01±0.00	38.14±0.00	49.93±0.00	46.59±0.00	1.14±0.00	32.55±0.00
	ROME	0.91±0.36	0.57±0.20	0.73±0.14	0.74±0.20	19.13±0.83	16.95±0.71	1.36±0.17	12.48±0.54
	AlphaEdit	64.90±0.41	31.19±0.80	4.86±0.04	33.65±0.39	63.83±0.23	56.93±0.02	21.12±0.14	47.29±0.06
	SUIT	30.29±0.09	10.65±0.44	4.81±0.17	15.25±0.12	43.90±0.15	38.69±0.12	18.09±0.08	33.56±0.11
	MEMIT	46.36±0.05	25.36±0.21	9.93±0.05	27.22±0.07	62.59±0.09	58.15±0.03	25.87±0.08	48.87±0.04
	MEMIT-CFPP	62.04±0.31	34.18±0.20	8.24±0.08	**34.82±0.09**	70.51±0.09	64.69±0.13	24.33±0.03	**53.18±0.08**
	PMET	18.63±0.49	8.81±0.31	10.77±0.06	12.74±0.24	38.63±0.68	35.93±0.57	25.07±0.16	33.21±0.37
	PMET-CFPP	26.03±0.13	13.15±0.15	9.76±0.08	**16.31±0.07**	41.57±0.06	38.20±0.08	25.03±0.08	**34.93±0.02**
	EAMET	55.83±0.11	34.29±0.10	7.45±0.02	32.52±0.03	60.18±0.09	57.09±0.01	24.76±0.09	47.34±0.05
	EAMET-CFPP	64.82±0.16	39.29±0.25	5.98±0.13	**36.70±0.06**	62.32±0.15	58.77±0.14	22.35±0.06	**47.81±0.08**
Mistral-7B	FT	72.74±0.00	49.01±0.00	2.75±0.00	41.50±0.00	52.68±0.00	52.84±0.00	10.30±0.00	38.61±0.00
	ROME	1.23±0.64	0.83±0.49	0.62±0.55	0.89±0.56	0.43±0.24	0.41±0.25	0.20±0.09	0.35±0.18
	AlphaEdit	86.74±0.08	43.47±0.28	21.44±0.02	50.55±0.11	88.63±0.06	82.78±0.17	48.85±0.05	73.42±0.07
	SUIT	87.74±0.33	44.78±1.59	18.26±0.05	50.26±0.57	89.04±0.06	85.65±0.00	47.74±0.06	74.14±0.01
	MEMIT	87.15±0.16	44.25±0.51	20.89±0.06	50.76±0.13	88.51±0.02	82.70±0.04	48.90±0.05	73.37±0.04
	MEMIT-CFPP	93.17±0.03	51.15±0.28	20.07±0.05	**54.80±0.11**	91.30±0.09	86.14±0.13	48.75±0.01	**75.39±0.07**
	PMET	85.25±0.23	49.50±0.28	16.82±0.06	50.53±0.04	85.84±0.04	82.10±0.03	48.99±0.03	72.31±0.01
	PMET-CFPP	86.57±0.21	51.71±0.33	15.97±0.04	**51.42±0.07**	86.06±0.05	82.45±0.04	48.78±0.03	**72.43±0.03**
	EAMET	78.38±0.08	47.69±0.06	16.27±0.04	47.45±0.02	87.51±0.04	85.48±0.01	47.67±0.01	73.55±0.01
	EAMET-CFPP	80.87±0.08	49.79±0.18	15.91±0.07	**48.86±0.01**	87.90±0.08	85.73±0.06	47.21±0.02	**73.62±0.03**
LLaMA3-8B	FT	98.99±0.00	74.09±0.00	5.51±0.00	59.53±0.00	67.90±0.00	67.95±0.00	31.42±0.00	55.76±0.00
	ROME	0.39±0.25	0.40±0.30	0.38±0.35	0.39±0.30	1.62±0.39	1.51±0.29	0.30±0.16	1.14±0.27
	AlphaEdit	75.26±0.87	37.25±0.33	22.05±0.11	44.85±0.21	81.55±0.03	77.80±0.01	41.42±0.13	66.92±0.03
	SUIT	90.53±0.93	51.87±2.67	16.97±0.13	53.13±1.18	85.07±0.05	81.86±0.04	40.86±0.10	69.26±0.02
	MEMIT	85.34±0.14	42.78±0.79	20.84±0.06	49.65±0.23	84.21±0.09	80.47±0.09	41.31±0.17	68.66±0.11
	MEMIT-CFPP	93.53±0.16	50.36±0.21	19.82±0.09	**54.57±0.09**	88.13±0.04	84.25±0.11	40.66±0.32	**71.01±0.14**
	PMET	82.38±0.40	41.88±0.07	17.48±0.03	47.25±0.14	82.63±0.10	78.94±0.03	41.38±0.04	67.65±0.06
	PMET-CFPP	85.32±0.42	44.69±0.20	16.60±0.10	**48.87±0.11**	83.25±0.14	79.62±0.14	41.16±0.12	**68.01±0.06**
	EAMET	76.89±0.58	37.92±0.51	21.10±0.17	45.30±0.03	85.50±0.07	82.88±0.08	43.11±0.05	70.50±0.03
	EAMET-CFPP	89.70±0.17	47.95±0.25	19.94±0.02	**52.53±0.10**	88.05±0.02	85.19±0.05	42.03±0.05	**71.76±0.02**

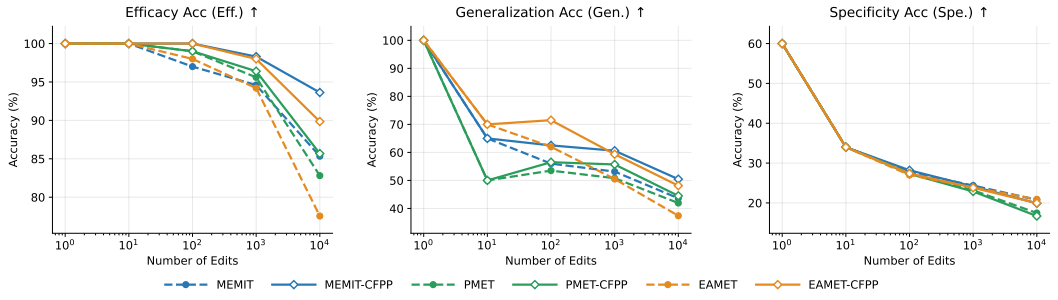


Figure 5: Scaling behavior on LLaMA3-8B for CounterFact.

The gains also transfer to other representative locate-then-edit methods: both PMET-CFPP and EAMET-CFPP consistently improve the Avg. score over their respective baselines across all three models and both datasets. This consistency suggests that CFPP is not tied to the MEMIT update rule: it modifies only the preservation covariance while leaving the underlying method's target-vector computation unchanged.

Figure 5 shows the CounterFact scaling behavior on LLaMA3-8B as the number of edits increases from 1 to 10^4. Across MEMIT, PMET, and EAMET, the CFPP variants generally retain higher efficacy or generalization at larger editing scales, while specificity remains competitive with the corresponding baselines. Additional scaling results for other model–dataset combinations are provided in Appendix D.7.

Table 4: Performance on Wiki-recent using all 1,266 edit requests in a smaller-scale evaluation.

Method	GPT2-XL				Mistral-7B				LLaMA3-8B			
	Eff.↑	Por.↑	Loc.↑	Avg.↑	Eff.↑	Por.↑	Loc.↑	Avg.↑	Eff.↑	Por.↑	Loc.↑	Avg.↑
MEMIT	47.23	13.87	18.33	26.48	84.81	22.32	29.33	45.48	85.65	26.25	31.74	47.88
MEMIT-CFPP	**67.96**	**16.84**	**19.70**	**34.83**	**92.92**	**24.08**	**30.31**	**49.10**	**94.72**	**27.60**	**33.84**	**52.05**
PMET	27.78	9.92	13.18	16.96	73.57	22.80	31.69	42.69	80.75	26.79	29.31	45.61
PMET-CFPP	**30.60**	**11.11**	12.47	**18.06**	**74.41**	**23.80**	**31.95**	**43.39**	**82.62**	**27.54**	**30.63**	**46.93**
EAMET	46.13	11.61	12.58	23.44	72.01	25.22	34.11	43.78	78.28	26.08	33.55	45.97
EAMET-CFPP	**54.35**	**13.77**	**13.72**	**27.28**	**73.78**	**25.40**	**34.41**	**44.53**	**86.33**	**28.11**	**35.94**	**50.13**

Compared with the commonly used CounterFact and ZsRE benchmarks, Wiki-recent comprises fine-grained Wikidata triplets representing recent factual updates, providing a temporally newer and time-grounded evaluation setting (Zhang et al., 2024). For models pretrained before these insertions, it also lowers the chance that the evaluated facts appeared in pretraining data. Table 4 evaluates the same methods on all 1,266 Wiki-recent requests. Additional baseline results are reported in Appendix D.5. CFPP improves the Avg. score in all paired comparisons and generally raises efficacy and portability while maintaining comparable or better locality. The largest relative gain is 31.5%, achieved by MEMIT-CFPP on GPT2-XL. These results show that CFPP remains effective at Wiki-recent's slightly smaller editing scale.

5.4 Sequential Editing Robustness (RQ3)

Sequential editing. To assess whether CFPP remains effective under sequential editing, we divide the requests into batches and update the model parameters after each batch. For a fair comparison, the MEMIT and MEMIT-CFPP entries in Table 5 denote their sequential variants: each batch is edited using the model produced by the preceding batches while retaining the constraints induced by earlier edits. Relative to sequential MEMIT, MEMIT-CFPP improves the Avg. score in all four dataset–scale comparisons. On CounterFact, it achieves the highest Avg. score at both scales, surpassing both AlphaEdit and SUIT; on ZsRE, it remains close to AlphaEdit and outperforms the sequential MEMIT baseline at both scales. These results show that CFPP retains its benefits after model parameters are updated across successive batches.

Table 5: Sequential editing results on LLaMA3-8B across CounterFact and ZsRE under the $1{,}000/100$ and $2{,}000/100$ settings. Best values are **bold**; second-best values are underlined.

Method	1,000 / 100								2,000 / 100							
	CounterFact				ZsRE				CounterFact				ZsRE			
	Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Gen.↑	Spe.↑	Avg.↑
MEMIT	97.10	57.75	**23.33**	59.39	89.16	83.53	39.58	70.76	96.50	59.68	**22.64**	59.61	90.59	84.80	40.22	71.87
AlphaEdit	96.60	61.20	21.59	59.80	89.99	84.93	**40.75**	71.89	96.80	64.03	18.25	59.69	91.51	**86.20**	**42.09**	73.27
SUIT	**99.00**	63.05	22.51	61.52	**92.55**	**86.99**	40.30	**73.28**	**99.00**	61.50	22.08	60.86	**93.41**	85.91	40.98	**73.43**
MEMIT-CFPP	97.40	**64.50**	23.29	**61.73**	90.54	84.97	39.40	71.64	96.50	**65.48**	22.18	**61.39**	91.94	85.95	39.94	72.61

6 Analysis

Our experiments suggest that the preservation–edit conflict is not solely inherent to model editing. Instead, it can arise from a geometric mismatch within the local closed-form objective. This perspective treats preservation not merely as a regularization strength to tune, but as a geometric constraint that should be compatible with the edit objective. CFPP further indicates that preservation geometry and projection strength should be controlled separately: spectral whitening makes directions comparable, adaptive gating adjusts projection strength according to the predicted edit effect. CFPP therefore complements prior work by shaping the preservation covariance before computing the local closed-form update. As long as future locate-then-edit methods rely on a preservation covariance, similar preservation–edit conflict may reappear and cause interference.

6.1 Ablation Study

To isolate the contributions of spectral whitening and residual-aware adaptive soft projection, we conduct an ablation on LLaMA3-8B across CounterFact, ZsRE, and Wiki-recent. The "spectral" and "adaptive" settings in Table 6 refer to the spectral-whitening and residual-aware adaptive soft projection components, respectively. The w/o adaptive variant replaces the directional gate with a uniform projection strength $q_i = \beta$ (see Appendix D.4), while w/o spectral disables spectral whitening. Trace calibration is enabled in all three CFPP settings.

Table 6: Ablation study of CFPP on LLaMA3-8B, isolating the effects of spectral whitening and residual-aware adaptive projection across CounterFact, ZsRE, and Wiki-recent.

Method	CounterFact				ZsRE				Wiki-recent			
	Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Gen.↑	Spe.↑	Avg.↑	Eff.↑	Por.↑	Loc.↑	Avg.↑
MEMIT	85.34	42.78	20.84	49.65	84.21	80.47	41.31	68.66	85.65	26.25	31.74	47.88
MEMIT-CFPP	93.53	**50.36**	19.82	**54.57**	**88.13**	**84.25**	40.66	**71.01**	94.72	27.60	33.84	**52.05**
w/o adaptive	**94.30**	49.94	19.31	54.52	87.45	83.26	40.87	70.53	**95.18**	27.38	33.47	52.01
w/o spectral	89.26	49.16	**20.93**	53.12	85.80	81.60	**41.04**	69.48	89.26	**28.45**	**35.64**	51.12

The complete MEMIT-CFPP configuration achieves the highest Avg. score on all three benchmarks, improving over MEMIT by 4.92, 2.35, and 4.25 percentage points on CounterFact, ZsRE, and Wiki-recent, respectively. Removing adaptive gating causes a small Avg. decrease on CounterFact and Wiki-recent but a larger one on ZsRE, whereas removing spectral whitening produces larger drops across all three benchmarks. These results suggest that spectral whitening drives the main gains in efficacy and generalization, while adaptive gating complements it by improving the editing–preservation balance.

6.2 Editing Efficiency

Figure 6 compares the edit time of MEMIT-CFPP with MEMIT, AlphaEdit, and SUIT at 1,000 and 10,000 edits. At both scales, MEMIT-CFPP is only slightly slower than MEMIT while remaining substantially faster than AlphaEdit and SUIT. This favorable runtime profile reflects the lightweight nature of CFPP: it modifies the preservation covariance before the existing local solve, without introducing retraining or an additional optimization stage. The consistent ordering across editing scales further suggests that the geometric preprocessing does not dominate the overall editing workload. The memory comparison in Appendix D.2 shows average usage close to MEMIT and maximum usage no larger at the 10,000-edit scale. Together, these results indicate that CFPP improves editing quality while preserving a practical computational and memory profile.

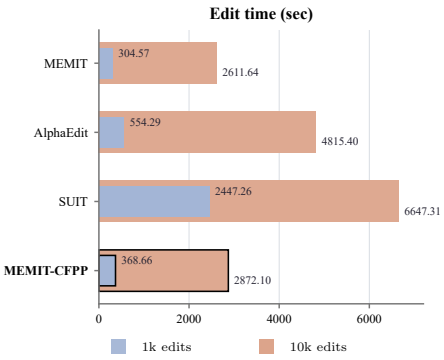


Figure 6: Edit time on LLaMA3-8B.

7 Conclusion

We introduced CFPP, a lightweight extension for locate-then-edit model editing. By modifying the preservation covariance before the local closed-form update, CFPP mitigates preservation–edit conflict through spectral whitening, residual-aware adaptive soft projection, and trace calibration, without retraining or trainable parameters. Across models and benchmarks, CFPP consistently improves editing performance under massive and sequential editing, transfers to representative locate-then-edit methods, and maintains competitive specificity or locality. Overall, these findings underscore the role of preservation geometry in reliable large-scale editing.

AI USE STATEMENT

During the preparation of this manuscript, we used generative AI tools to assist with language editing, proofreading, and improving the clarity and consistency of the written presentation. Generative AI was also used to support selected mathematical proofs in the appendix, including checking algebraic steps and clarifying their exposition. We independently reviewed and verified all AI-assisted text and proofs and take full responsibility for the final content of this work.

ETHICS STATEMENT

Knowledge editing can improve the reliability of large language models but may also introduce risks, including misinformation, biased outputs, and unintended behavioral changes if used carelessly. Our experiments rely on publicly available code, datasets, and models and follow their stated licenses and terms of use. We recommend validating edited facts, assessing side effects, and monitoring model behavior in controlled deployments.

REPRODUCIBILITY STATEMENT

All experiments in this paper are reproducible. The code and datasets are publicly available at the following anonymous repository: `https://anonymous.4open.science/r/CFPP-B4H7`.

REFERENCES

Yuchen Cai and Ding Cao. O-Edit: Orthogonal subspace editing for language model sequential editing. *CoRR*, abs/2410.11469, 2024. doi: 10.48550/ARXIV.2410.11469. URL https://doi.org/10.48550/arXiv.2410.11469.

Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 6491–6506. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.522. URL https://doi.org/10.18653/v1/2021.emnlp-main.522.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 8493–8502. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.ACL-LONG.581. URL https://doi.org/10.18653/v1/2022.acl-long.581.

Yanbo Dai, Zhenlan Ji, Zongjie Li, and Shuai Wang. EAMET: Robust massive model editing via embedding alignment optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VJGVfJJ2wW.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=HvSytvg3Jh.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 5484–5495. Association for Computational Linguistics, 2021. doi: 10.18653/V1/2021.EMNLP-MAIN.446. URL https://doi.org/10.18653/v1/2021.emnlp-main.446.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods*

in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022, pp. 30–45. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.3. URL https://doi.org/10.18653/v1/2022.emnlp-main.3.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. Model editing at scale leads to gradual and catastrophic forgetting. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, Findings of ACL, pp. 15202–15232. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.902. URL https://doi.org/10.18653/v1/2024.findings-acl.902.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with GRACE: lifelong model editing with discrete key-value adaptors. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/95b6e2ff961580e03c0a662a63a71812-Abstract-Conference.html.

Chenhui Hu, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Knowledge in superposition: Unveiling the failures of lifelong knowledge editing for large language models. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 24086–24094. AAAI Press, 2025. doi: 10.1609/AAAI.V39I22.34583. URL https://doi.org/10.1609/aaai.v39i22.34583.

Houcheng Jiang, Junfeng Fang, Ningyu Zhang, Mingyang Wan, Guojun Ma, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AnyEdit: Edit any knowledge encoded in language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025a. URL https://proceedings.mlr.press/v267/jiang25b.html.

Houcheng Jiang, Junfeng Fang, Tianyu Zhang, Baolong Bi, An Zhang, Ruipeng Wang, Tao Liang, and Xiang Wang. Neuron-level sequential editing for large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 16678–16702. Association for Computational Linguistics, 2025b. URL https://aclanthology.org/2025.acl-long.815/.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Trans. Assoc. Comput. Linguistics*, 9:962–977, 2021. doi: 10.1162/TACL_A_00407. URL https://doi.org/10.1162/tacl_a_00407.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In Roger Levy and Lucia Specia (eds.), *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pp. 333–342. Association for Computational Linguistics, 2017. doi: 10.18653/V1/K17-1034. URL https://doi.org/10.18653/v1/K17-1034.

Qi Li and Xiaowen Chu. AdaEdit: Advancing continuous knowledge editing for large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 4127–4149. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.208/.

Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. PMET: precise model editing in a transformer. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (eds.),

Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2024, February 20-27, 2024, Vancouver, Canada, pp. 18564–18572. AAAI Press, 2024. doi: 10.1609/AAAI.V38I17.29818. URL https://doi.org/10.1609/aaai.v38i17.29818.

Sicheng Lyu, Yu Gu, Xinyu Wang, Jerry Huang, Sitao Luan, Yufei Cui, Xiao-Wen Chang, and Peng Lu. EvoEdit: Evolving null-space alignment for robust and efficient knowledge editing. *CoRR*, abs/2510.13851, 2025. doi: 10.48550/ARXIV.2510.13851. URL https://doi.org/10.48550/arXiv.2510.13851.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022a. URL https://openreview.net/forum?id=0DcZxeWfOPt.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, Proceedings of Machine Learning Research, pp. 15817–15831. PMLR, 2022b. URL https://proceedings.mlr.press/v162/mitchell22a.html.

Haewon Park, Sangwoo Kim, and Yohan Jo. SUIT: Knowledge editing with subspace-aware key-value mappings. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=qz3BkyyHWJ.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. Language models as knowledge bases? In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 2463–2473. Association for Computational Linguistics, 2019. doi: 10.18653/V1/D19-1250. URL https://doi.org/10.18653/v1/D19-1250.

Siyuan Qi, Bangcheng Yang, Kailin Jiang, Xiaobo Wang, Jiaqi Li, Yifan Zhong, Yaodong Yang, and Zilong Zheng. In-context editing: Learning knowledge from self-induced distributions. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=w6rHCuN3YG.

Shanbao Qiao, Xuebing Liu, and Seung-Hoon Na. Wasserstein distance constraint and parameter sparsification for batched and iterative knowledge editing. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 25019–25028. AAAI Press, 2025. doi: 10.1609/AAAI.V39I23.34686. URL https://doi.org/10.1609/aaai.v39i23.34686.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing,*

EMNLP 2020, Online, November 16-20, 2020, pp. 5418–5426. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.437. URL https://doi.org/10.18653/v1/2020.emnlp-main.437.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 4222–4235. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.346. URL https://doi.org/10.18653/v1/2020.emnlp-main.346.

Wei Sun, Tingyu Qu, Mingxiao Li, Jesse Davis, and Marie-Francine Moens. Mitigating negative interference in multilingual knowledge editing through null-space constraints. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Findings of ACL, pp. 8796–8810. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.findings-acl.460/.

Haoyu Xu, Pengxiang Lan, Enneng Yang, Guibing Guo, Jianzhe Zhao, Linying Jiang, and Xingwei Wang. Knowledge decoupling via orthogonal projection for lifelong editing of large language models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 13194–13213. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.646/.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. Editing large language models: Problems, methods, and opportunities. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 10222–10240. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.632. URL https://doi.org/10.18653/v1/2023.emnlp-main.632.

Mengqi Zhang, Xiaotian Ye, Qiang Liu, Shu Wu, Pengjie Ren, and Zhumin Chen. Uncovering overfitting in large language model editing. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=t8qcGXaepr.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. A comprehensive study of knowledge editing for large language models. *CoRR*, abs/2401.01286, 2024. doi: 10.48550/ARXIV.2401.01286. URL https://doi.org/10.48550/arXiv.2401.01286.

A Subspace Diagnostics and Analysis

A.1 Subspace Conflict Diagnostics

The preservation covariance C_p is constructed from preserved keys K_p. Its top eigenvectors identify the dominant directions along which the preservation term imposes the strongest penalty. In contrast, the left singular vectors of K_e span the principal directions of the active edit subspace. Thus, comparing the top eigenspace of C_p with the leading left singular subspace of K_e measures whether the directions most strongly protected by preservation overlap with the directions required by editing.

Principal angles quantify the geometric separation between two subspaces. If the singular values of $U_p^\top U_e$ are large, the corresponding principal angles are small, meaning that the two subspaces contain nearly aligned directions. In our setting, such alignment indicates that the preservation term is likely to penalize directions used by the edit update. Conversely, larger principal angles indicate that the preservation and edit subspaces are more separated, implying weaker geometric conflict.

The normalized Frobenius overlap score aggregates this alignment across all selected directions:

$$\mathrm{Overlap}(U_p, U_e) = \frac{\|U_p^\top U_e\|_F^2}{r} = \frac{1}{r}\sum_{i=1}^{r} \cos^2 \theta_i. \tag{13}$$

This value lies in $[0, 1]$ when U_p and U_e have orthonormal columns and the same rank. It equals 0 when the two subspaces are orthogonal and equals 1 when they coincide. Therefore, the overlap score gives a compact scalar summary of how much the dominant preservation subspace induced by C_p intersects with the edit subspace induced by K_e. A larger score indicates stronger preservation–edit conflict, while a smaller score indicates that the preservation term is less likely to suppress edit-relevant directions.

A.2 Orthogonal Preservation under Projection

The following hard-projector argument isolates the orthogonal-component property underlying the soft covariance projection used by CFPP. Let $P_e = K_e(K_e^\top K_e)^+ K_e^\top$ be the orthogonal projector onto $\mathrm{span}(K_e)$, and let $P_{\perp e} = I - P_e$ be the projector onto its orthogonal complement. For any preserved key k_p, decompose it as:

$$k_p = k_p^{\parallel} + k_p^{\perp}, \tag{14}$$

where $k_p^{\parallel} \in \mathrm{span}(K_e)$ and $k_p^{\perp} \in \mathrm{span}(K_e)^{\perp}$. By the properties of orthogonal projection,

$$P_{\perp e} k_p = P_{\perp e} k_p^{\parallel} + P_{\perp e} k_p^{\perp} = 0 + k_p^{\perp} = k_p^{\perp}. \tag{15}$$

Under this hard projection, the component orthogonal to $\mathrm{span}(K_e)$ is preserved exactly, while the component aligned with the edit subspace is removed. The soft projector used by CFPP analogously leaves the complementary directions unchanged in whitened coordinates while attenuating the selected edit-aligned components according to Q. Applying the hard projection column-wise to the preserved keys K_p gives

$$P_{\perp e} K_p = K_p^{\perp}. \tag{16}$$

Therefore, the projected preservation covariance becomes

$$\begin{aligned} P_{\perp e} K_p K_p^\top P_{\perp e} &= (P_{\perp e} K_p)(P_{\perp e} K_p)^\top \\ &= K_p^{\perp} (K_p^{\perp})^\top. \end{aligned} \tag{17}$$

This hard-projection identity explains the orthogonal-component property underlying CFPP. The same argument extends to the whitened operator under the Mahalanobis inner product $\langle x, y\rangle_{C_\lambda^{-1}} \triangleq x^\top C_\lambda^{-1} y$, where $C_\lambda = LL^\top \succ 0$. Since $C_\lambda^{-1} = L^{-\top} L^{-1}$, whitening is an isometry because $\langle x, y\rangle_{C_\lambda^{-1}} = (L^{-1}x)^\top (L^{-1}y)$. For the retained basis U_r, let $\bar{P}_{\perp r} = I - U_r U_r^\top$ and define its pullback as

$$P_{\perp r}^{(\lambda)} = L \bar{P}_{\perp r} L^{-1}, \qquad \big(P_{\perp r}^{(\lambda)}\big)^2 = P_{\perp r}^{(\lambda)}, \qquad \big(P_{\perp r}^{(\lambda)}\big)^\top C_\lambda^{-1} = C_\lambda^{-1} P_{\perp r}^{(\lambda)}. \tag{18}$$

For any preserved key, let $k_p^{\parallel_\lambda} = L U_r U_r^\top L^{-1} k_p$ and $k_p^{\perp_\lambda} = P_{\perp r}^{(\lambda)} k_p$. Then

$$k_p = k_p^{\parallel_\lambda} + k_p^{\perp_\lambda}, \qquad \langle k_p^{\perp_\lambda}, L U_r a\rangle_{C_\lambda^{-1}} = 0, \qquad P_{\perp r}^{(\lambda)} k_p^{\perp_\lambda} = k_p^{\perp_\lambda}, \quad P_{\perp r}^{(\lambda)} k_p^{\parallel_\lambda} = 0. \tag{19}$$

Thus, the component orthogonal to the retained edit subspace is preserved exactly. Moreover,

$$\begin{aligned} P_{\perp r}^{(\lambda)} C_\lambda \big(P_{\perp r}^{(\lambda)}\big)^\top &= L \bar{P}_{\perp r} L^\top, \\ \big(L P_q L^{-1}\big) C_\lambda \big(L P_q L^{-1}\big)^\top &= L P_q P_q^\top L^\top = \widetilde{C}_q, \end{aligned} \tag{20}$$

where $P_q = I - U_r Q U_r^\top$ is the soft counterpart. It is the identity on the complementary whitened directions and maps u_i to $(1 - q_i)u_i$. Hence, for $0 < q_i < 1$, it attenuates edit-aligned directions rather than removing them.

B Implementation Details

B.1 Rank Truncation

We support restricting the projection to the dominant edit directions. In the default spectral path, we compute a truncated basis from the leading singular directions of $\bar{K}_e = L^{-1}K_e$ and construct the projector on this reduced whitened subspace; the non-spectral path applies the analogous truncation directly to K_e. This option is controlled by `cfpp_rank`, which specifies the number or fraction of retained edit directions. Here, r denotes the effective retained rank. When `cfpp_rank` is unset, CFPP retains all numerically nonzero edit directions. Setting `cfpp_rank=r` retains the top-r directions. Percentage specifications are also supported, allowing the retained rank to scale with the number of edits. This design makes CFPP more robust as the editing scale increases, because the raw edit-key matrix can contain weak or noisy directions that should not necessarily be projected out of the preservation term.

B.2 Spectral and Budget Stabilization

The released implementation applies the covariance transform before each method's local closed-form update. For every selected layer and rewritten module, it first loads the corresponding preservation covariance C_p, symmetrizes it, and constructs C_λ in Equation 6. The initial ridge is scaled by the mean absolute diagonal of the empirical covariance rather than by a fixed dimensional constant. If the empirical statistic is numerically indefinite, the ridge is increased geometrically until Cholesky factorization succeeds. This treatment is important for large models because the whitening step is performed in float32 to limit peak memory, while the subsequent computation of the local closed-form solution is carried out in double precision.

For a batch containing N_e edits, the implementation obtains the spectral basis without decomposing the full hidden dimension. Specifically, after whitening, it first constructs the edit Gram matrix:

$$G_e = \bar{K}_e^\top \bar{K}_e. \tag{21}$$

We then compute the leading eigensystem of this small matrix:

$$G_e V_r = V_r \Sigma_r^2. \tag{22}$$

Finally, the corresponding left singular vectors are recovered by

$$U_r = \bar{K}_e V_r \Sigma_r^{-1}. \tag{23}$$

The implementation retains the corresponding singular directions only when $\sigma_i > \texttt{cfpp_rtol}\,\sigma_1$, where σ_1 is the largest singular value. We set $\texttt{cfpp_rtol} = 10^{-6}$ in all experiments, so directions with $\sigma_i \le 10^{-6}\sigma_1$ are treated as numerically negligible. The retained rank is then bounded by the configured rank. Thus, the eigendecomposition scales with the number of requests rather than the full representation dimension. Percentage-style ranks use $\min(N_e, d_k)$ as their reference size, preventing an edit batch larger than the hidden dimension from implicitly forcing a full-rank projector.

B.3 Trace Calibration

In the default spectral path, the low-rank covariance correction can reduce the trace of the projected covariance relative to the stabilized preservation covariance C_λ defined in Equation 6. The ridge is chosen according to the diagonal scale of C_p. We set $\epsilon_{\text{sp}} = 10^{-6}$ in all experiments. This choice makes δ typically small relative to the scale of C_p; it may be increased when stronger regularization is required for numerical stability. When C_p is numerically positive semidefinite, $\operatorname{tr}(C_\lambda) = (1 + \epsilon_{\text{sp}})\operatorname{tr}(C_p)$, so the two trace budgets are nearly identical.

To preserve this stabilized covariance budget, we rescale $\widetilde{C}_q$ as follows:

$$\alpha = \frac{\operatorname{tr}(C_\lambda)}{\operatorname{tr}(\widetilde{C}_q)}, \qquad C_{\text{CFPP}} = \alpha \widetilde{C}_q. \tag{24}$$

This calibration preserves the total stabilized second-moment budget while attenuating the edit-aligned penalty. Accordingly, in the actual implementation, the preservation covariance used in Equation 11 is C_{CFPP}, obtained by replacing $\widetilde{C}_q$ with $\alpha\widetilde{C}_q$ after trace calibration.

C EVALUATION SETUP

C.1 METRICS

Existing model-editing work uses two related families of evaluation metrics. The first is a *probability-based success* criterion, which checks whether the edited target becomes more likely than the original target. The second is an *accuracy-based* criterion, which checks whether the model explicitly predicts the desired answer. Our main editing metrics use accuracy-based definitions.

Probability-based success metrics. Let x_i^{eff}, x_i^{gen}, x_i^{por}, x_i^{spe}, and x_i^{loc} denote efficacy, generalization, portability, specificity, and locality prompts, respectively, when available for a given benchmark. Let $o_i^\star$ denote the edited target answer and o_i^{old} the original answer before editing. Probability-based success metrics compare the likelihoods of these two answers: Editing efficacy measures whether the edited target becomes more likely than the original answer on the rewrite prompt:

$$\text{Succ}_{\text{eff}} = \mathbb{E}_i\left[\mathbb{P}_{f_\theta}(o_i^\star \mid x_i^{\text{eff}}) > \mathbb{P}_{f_\theta}(o_i^{\text{old}} \mid x_i^{\text{eff}})\right]. \tag{25}$$

Generalization measures the same probability comparison on paraphrase prompts:

$$\text{Succ}_{\text{gen}} = \mathbb{E}_i\left[\mathbb{P}_{f_\theta}(o_i^\star \mid x_i^{\text{gen}}) > \mathbb{P}_{f_\theta}(o_i^{\text{old}} \mid x_i^{\text{gen}})\right]. \tag{26}$$

Portability uses the same definition with x_i^{por} in place of x_i^{gen}. Specificity measures whether the original answer remains more likely on neighborhood prompts:

$$\text{Succ}_{\text{spe}} = \mathbb{E}_i\left[\mathbb{P}_{f_\theta}(o_i^{\text{old}} \mid x_i^{\text{spe}}) > \mathbb{P}_{f_\theta}(o_i^\star \mid x_i^{\text{spe}})\right]. \tag{27}$$

Locality uses the same definition with x_i^{loc} in place of x_i^{spe}. These probability-comparison metrics are widely used in prior work.

Accuracy-based metrics. In the main paper, however, we report accuracy-style metrics. For CounterFact and Wiki-recent, the metrics use strict token-level exact-match evaluation of the target answer: Editing efficacy is the strict top-1 accuracy of the edited target on the rewrite prompt:

$$\text{Acc}_{\text{eff}} = \mathbb{E}_i\left[o_i^\star = \arg\max_o \mathbb{P}_{f_\theta}(o \mid x_i^{\text{eff}})\right]. \tag{28}$$

Generalization is the strict top-1 accuracy on paraphrase prompts:

$$\text{Acc}_{\text{gen}} = \mathbb{E}_i\left[o_i^\star = \arg\max_o \mathbb{P}_{f_\theta}(o \mid x_i^{\text{gen}})\right]. \tag{29}$$

Portability uses the same definition with x_i^{por} in place of x_i^{gen}. Specificity is the strict top-1 accuracy of the original answer on neighborhood prompts:

$$\text{Acc}_{\text{spe}} = \mathbb{E}_i\left[o_i^{\text{old}} = \arg\max_o \mathbb{P}_{f_\theta}(o \mid x_i^{\text{spe}})\right]. \tag{30}$$

Locality uses the same definition with x_i^{loc} in place of x_i^{spe}. Only the prompt types available for each benchmark are instantiated. ZsRE uses the token-level definition below.

Generation fluency. For a generated continuation j, let $g_{n,j}(k)$ denote the empirical probability of the k-th n-gram. We compute its entropy as $H_{n,j} = -\sum_k g_{n,j}(k)\log_2 g_{n,j}(k)$ and report the mean weighted bigram–trigram entropy:

$$\text{Flu.} = \frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}\left(\frac{2}{3}H_{2,j} + \frac{4}{3}H_{3,j}\right), \qquad H_{n,j} = -\sum_k g_{n,j}(k)\log_2 g_{n,j}(k). \tag{31}$$

Reference consistency. Let G be the concatenation of the generated continuations and R the concatenation of the relation-specific reference snippets. If $\boldsymbol{v}_G$ and $\boldsymbol{v}_R$ are their TF–IDF vectors under the fixed evaluation vectorizer, we compute reference consistency as their cosine similarity:

$$\text{Consis.} = \frac{\boldsymbol{v}_G^\top \boldsymbol{v}_R}{\|\boldsymbol{v}_G\|_2\,\|\boldsymbol{v}_R\|_2}. \tag{32}$$

C.2 DATASETS

We evaluate CFPP on three complementary knowledge-editing benchmarks. CounterFact and ZsRE provide established factual rewriting settings with paraphrase and preservation prompts, while Wiki-recent extends the evaluation to recent Wikidata-derived updates with portability and locality tests. Together, these benchmarks cover direct rewriting, paraphrase generalization, knowledge transfer, and preservation of unrelated facts.

CounterFact. This benchmark contains 21,919 factual-editing cases organized around subject–relation–answer tuples. Each case specifies an original answer and a counterfactual target answer, together with paraphrase prompts that preserve the edited fact under alternative wording and neighborhood prompts that query related but unedited facts. This structure supports separate evaluation of editing efficacy, paraphrase generalization, and preservation of nearby knowledge (Meng et al., 2022).

For CounterFact, we report **efficacy**, **generalization**, and **specificity** in accuracy form in the main tables. Efficacy measures whether the edited model predicts the target new answer on the original rewrite prompt. Generalization measures the same strict answer-level accuracy on paraphrase prompts associated with the edited fact. Specificity measures answer-level accuracy on neighborhood prompts that should still preserve the original ground-truth answer.

ZsRE. This question–answering benchmark is designed to evaluate zero-shot relation extraction. We use 19,086 cases, each centered on an original question and answer, accompanied by a paraphrased question with the same answer and an unrelated question for evaluating preservation of unrelated knowledge. Its question-based format enables editing success and transfer to be evaluated directly through the model's predicted answers (Levy et al., 2017).

For ZsRE, we also report **efficacy**, **generalization**, and **specificity**, but here each metric is computed as an average token-level hit rate. Let the target answer sequence be $(y^\star_{i,1}, \ldots, y^\star_{i,T_i})$. Then the token-level accuracy for a prompt x_i is

$$\mathrm{Acc}_{\mathrm{tok}}(x_i) = \frac{1}{T_i} \sum_{t=1}^{T_i} \left[y^\star_{i,t} = \arg\max_y \ \mathbb{P}_{f_\theta}(y \mid x_i, y^\star_{i,<t}) \right]. \tag{33}$$

For specificity, the original answer associated with each neighborhood prompt is used as the target, whereas efficacy and generalization use the edited target answer. The reported ZsRE efficacy, generalization, and specificity are obtained by averaging $\mathrm{Acc}_{\mathrm{tok}}(x_i)$ over rewrite, paraphrase, and neighborhood prompts, respectively. Thus, ZsRE remains accuracy-based, but it is measured at the token level rather than by strict whole-answer exact match.

Wiki-recent. The benchmark consists of 1,266 factual triplets recently incorporated into Wikidata. It is designed to simulate updating models with knowledge that emerged after their pretraining, providing a temporally grounded setting that reduces, but does not eliminate, the possibility of pretraining contamination. Editing methods are evaluated in terms of efficacy, portability, and locality. Portability measures whether the edited model can infer downstream consequences of the updated facts, whereas locality assesses whether unrelated knowledge is preserved (Zhang et al., 2024).

For Wiki-recent, we report **efficacy**, **portability**, and **locality**, again using accuracy-style evaluation. Efficacy measures whether the edited model retrieves the updated target fact on the primary rewrite prompt. Portability measures whether the same updated knowledge transfers correctly to reasoning variants or alias-based prompts associated with the edited fact. Locality measures whether unrelated prompts continue to produce their original correct answers after editing.

Dataset documentation. All three benchmarks are publicly available English-language knowledge-editing datasets. We use their original releases and task definitions without collecting additional data or introducing new annotations. All evaluations follow the prompt structures, answer annotations, and protocols provided by the respective benchmark releases.

D SUPPLEMENTARY EXPERIMENTAL RESULTS

D.1 RANK SELECTION

The overlap and principal-angle diagnostics depend on the rank r. We compute these diagnostics on CounterFact using 1,000 edit requests and the same final-layer selection as in the main analysis. Table 7 summarizes their behavior as r varies across GPT2-XL, Mistral-7B, and LLaMA3-8B. The diagnostics largely stabilize by $r = 32$: the overlap and θ_{mean} change only modestly beyond this point, while θ_{max} is already close to its ceiling. Increasing r mainly extends the tail of small principal angles, as reflected by a gradual decrease in θ_{min}, while adding computational cost. We therefore use $r = 32$ as a stable and cost-effective choice.

Table 7: Rank sweep of the preservation–edit conflict diagnostics.

Rank	GPT2-XL				Mistral-7B				LLaMA3-8B			
	Overlap	θ_{mean}	θ_{max}	θ_{min}	**Overlap**	θ_{mean}	θ_{max}	θ_{min}	**Overlap**	θ_{mean}	θ_{max}	θ_{min}
1	0.4095	50.21	50.21	50.21	0.0627	75.50	75.50	75.50	0.0481	77.33	77.33	77.33
2	0.2829	64.18	86.98	41.37	0.0466	80.62	89.00	72.24	0.0397	81.74	89.85	73.63
4	0.2189	67.25	88.15	34.91	0.0773	79.38	88.55	56.87	0.0725	79.69	89.88	58.30
8	0.1647	71.33	87.76	32.12	0.0615	80.58	89.79	51.27	0.0659	80.16	89.82	49.91
16	0.1400	73.44	89.82	28.67	0.0637	80.08	89.87	48.71	0.0607	79.60	89.35	47.43
32	0.1302	73.89	89.94	25.11	0.0640	79.46	89.92	44.08	0.0733	78.07	89.99	42.81
64	0.1174	74.83	89.99	19.44	0.0644	78.86	89.80	39.37	0.0792	77.41	89.92	36.94
128	0.1206	73.99	89.90	14.45	0.0798	77.46	89.98	32.50	0.0757	77.89	89.99	33.15

D.2 MEMORY USAGE

We provide the detailed memory comparison omitted from the main text for 1,000 and 10,000 edits on LLaMA3-8B. All measurements use float32 precision and were collected on NVIDIA RTX 6000D hardware with 84 GB of VRAM, using PyTorch 2.11.0+cu128 (CUDA 12.8 build). Figure 7 reports average and maximum memory usage in its two panels, respectively.

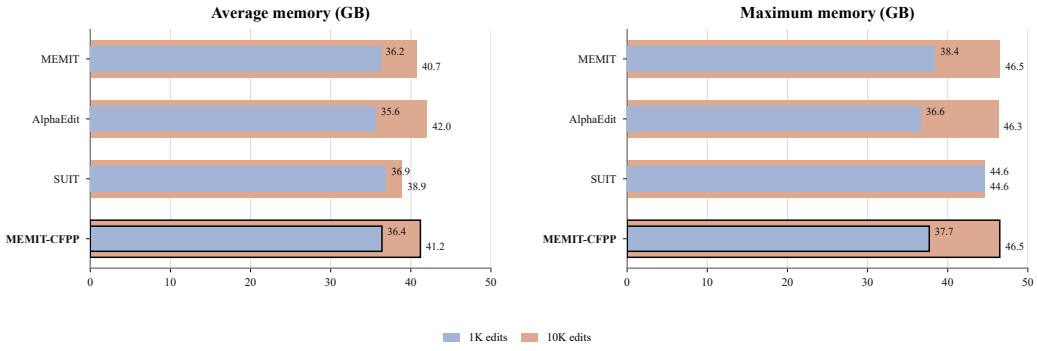


Figure 7: Average and maximum memory usage for massive editing on LLaMA3-8B.

Overall, CFPP has a memory profile comparable to MEMIT across both editing scales. Its average memory usage increases moderately as the number of edits grows, while remaining close to the baseline and below AlphaEdit at 10,000 edits. Peak memory is lower than MEMIT at 1,000 edits and reaches a similar level at 10,000 edits. These results indicate that CFPP does not introduce a disproportionate memory burden relative to existing locate-then-edit methods.

D.3 WIKI-RECENT CONFLICT ANALYSIS

We separately report the Wiki-recent conflict diagnostics in Table 8 using the same final-layer selection, rank-32 subspaces, and 1,000-request evaluation protocol as the analysis in the main paper.

Table 8: Conflict diagnostics on Wiki-recent for GPT2-XL, Mistral-7B, and LLaMA3-8B.

Model	Overlap↓		θ_{mean} ↑	
	Before CFPP	After CFPP	Before CFPP	After CFPP
GPT2-XL	0.1275	0.1047	74.21	76.03
Mistral-7B	0.0728	0.0639	78.90	79.72
LLaMA3-8B	0.0909	0.0829	76.47	77.18

Across all three models, CFPP reduces the normalized overlap and increases the mean principal angle. This consistent shift toward lower overlap and greater angular separation indicates that CFPP alleviates the preservation–edit conflict on Wiki-recent, consistent with the diagnostics reported in the main paper.

D.4 Gate Temperature Sensitivity

We further examine the sensitivity of the residual-aware adaptive gate to its temperature parameter τ, corresponding to `cfpp_gate_tau`. The results in Table 9 use MEMIT on LLaMA3-8B with 10,000 CounterFact edit requests. The first row is raw MEMIT without CFPP projection. Here, $\beta_{\text{median}} = \text{median}(q_1, \ldots, q_r)$ denotes the median gate value across the retained edit directions for each τ.

Table 9: Sensitivity to the adaptive gate temperature τ on LLaMA3-8B with MEMIT on CounterFact.

τ	β_{median}	Eff.↑	Gen.↑	Spe.↑	Avg.↑
MEMIT-RAW	0	85.34	43.63	**20.77**	49.91
39.00	0.05	87.55	43.70	20.74	50.66
19.00	0.10	88.63	44.84	20.54	51.34
9.00	0.20	90.65	47.66	20.35	52.89
5.00	0.333	92.10	49.18	20.10	53.79
3.00	0.500	93.63	50.45	19.91	54.66
2.00	0.667	94.49	51.72	19.55	55.25
1.50	0.800	94.89	51.52	19.17	55.19
1.25	0.889	**95.07**	**52.18**	19.16	**55.47**

Most experiments use $\tau = 3.0$. In uniform-projection ablations, we set β to the corresponding β_{median} to match the median projection strength. As τ decreases, the gate becomes stronger on typical edit directions and the Avg. score generally improves. Although $\tau = 1.25$ achieves the highest Avg. score (55.47), we use $\tau = 3.0$ as a more conservative operating point. It retains most of the Avg. improvement (54.66) while causing a smaller decrease in neighborhood specificity (19.91 versus 19.16), providing a better efficacy–specificity trade-off for the main experiments.

D.5 Wiki-recent Baseline Comparisons

For completeness, Table 10 reports FT, ROME, AlphaEdit, and SUIT on all 1,266 Wiki-recent requests. These results use the same evaluation protocol as Table 4 and provide additional baseline comparisons across the three models.

Table 10: Additional baseline results on Wiki-recent using all 1,266 requests.

Method	GPT2-XL				Mistral-7B				LLaMA3-8B			
	Eff.↑	Por.↑	Loc.↑	Avg.↑	Eff.↑	Por.↑	Loc.↑	Avg.↑	Eff.↑	Por.↑	Loc.↑	Avg.↑
FT	29.07	16.01	14.95	20.01	30.09	17.80	15.07	20.99	31.04	15.30	14.03	20.13
ROME	27.12	4.42	3.48	11.67	4.79	0.92	0.41	2.04	1.24	0.38	0.19	0.60
AlphaEdit	82.96	18.88	22.09	41.31	85.86	22.94	30.54	46.45	78.72	26.29	31.07	45.36
SUIT	84.81	15.91	21.41	40.71	85.97	20.74	26.16	44.29	80.04	27.19	31.27	46.17

D.6 Post-edit Generation Results

We additionally report two post-edit generation metrics after applying 10,000 edits on the CounterFact dataset, as shown in Table 11. Following the evaluation implementation, **Flu.** denotes the post-edit n-gram entropy, computed from a weighted combination of 2-gram and 3-gram entropies and used as a fluency proxy. **Consis.** denotes the post-edit reference score, computed as the TF–IDF cosine similarity between the generated texts and relation-specific reference snippets associated with the edited target.

Table 11: Mean post-edit generation metrics after 10,000 edits on the CounterFact dataset.

Method	GPT2-XL		Mistral-7B		LLaMA3-8B	
	Flu.↑	Consis.↑	Flu.↑	Consis.↑	Flu.↑	Consis.↑
MEMIT	624.93	36.30	616.78	31.80	631.54	29.48
MEMIT-CFPP	622.66	38.17	616.05	32.64	630.11	30.70
PMET	573.40	19.32	617.94	33.30	632.40	29.74
PMET-CFPP	626.59	31.31	617.93	33.77	632.16	30.30
EAMET	614.45	36.61	615.19	32.22	622.69	27.41
EAMET-CFPP	612.21	37.62	614.90	32.77	622.68	29.03

Across all paired comparisons, CFPP increases Consis., indicating stronger agreement between the generated text and the relation-specific reference snippets. Flu. remains close to the corresponding baseline in most comparisons, with only minor changes after 10,000 edits; the notable exception is PMET on GPT2-XL, where Flu. increases from 573.40 to 626.59. These results suggest that CFPP improves editing performance without causing a systematic degradation in generation fluency.

D.7 Cross-Model Scaling Results

To complement the main scaling analysis in Section 5.3, we report additional plots using the same evaluation protocol as Figure 5. These plots cover GPT2-XL and Mistral-7B on CounterFact and ZsRE, as well as LLaMA3-8B on ZsRE, with the number of edits increasing from 1 to 10^4. Across these supplementary settings, the CFPP-enhanced variants of MEMIT, PMET, and EAMET consistently preserve the qualitative trends observed in the main text, maintaining stronger efficacy or generalization at larger edit scales while keeping specificity competitive with the corresponding baselines.

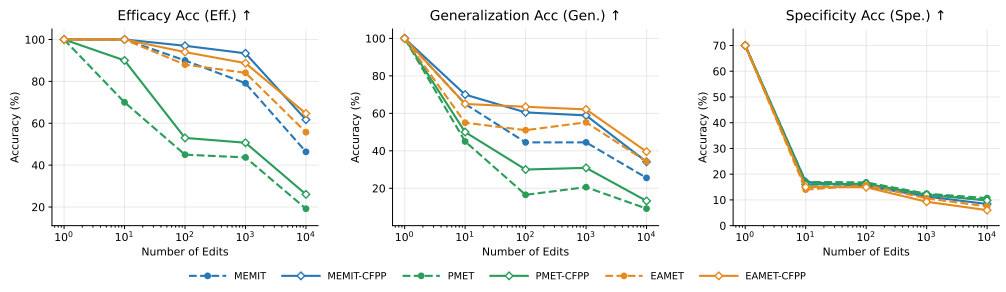


Figure 8: Scaling results for GPT2-XL on CounterFact.

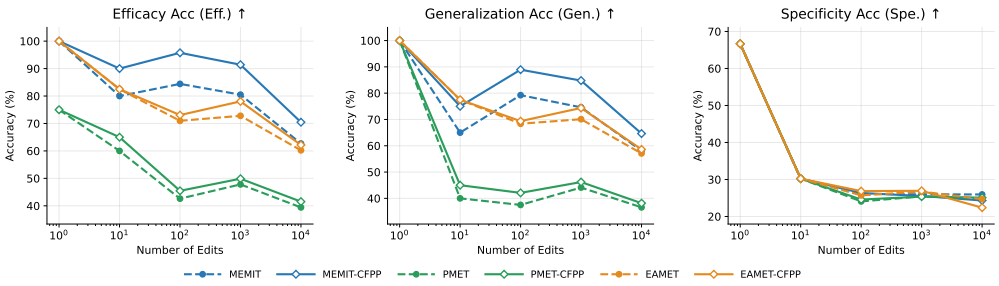


Figure 9: Scaling results for GPT2-XL on ZsRE.

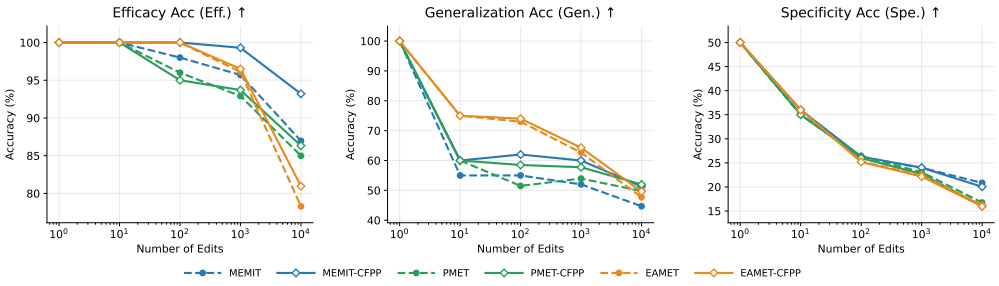


Figure 10: Scaling results for Mistral-7B on CounterFact.

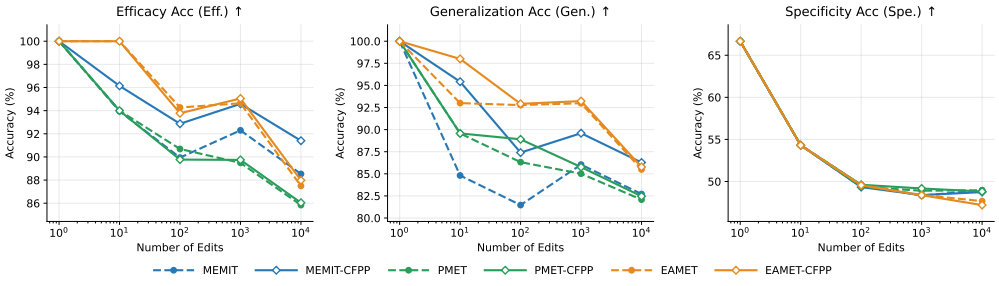


Figure 11: Scaling results for Mistral-7B on ZsRE.

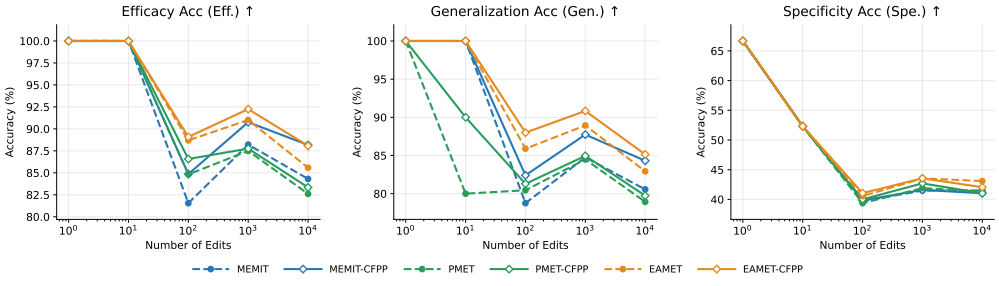


Figure 12: Scaling results for LLaMA3-8B on ZsRE.