ROSE: Risk-Aware Orthogonal Subspace Navigation for Lifelong Knowledge Editing in Multimodal Large Language Models

Lingyun Song[1,2]*, **Ziyao Chen**[1], **Kang Pan**[3], **Xiaolin Han**[1,*]
Yifei Zhang[1], **Xiaoqi Wang**[1], **Yudai Pan**[1,*], **Xuequn Shang**[1,4,*]

[1]School of Computer Science, Northwestern Polytechnical University, Xi'an
[2]Shenzhen Research Institute of Northwestern Polytechnical University, Shenzhen
[3]Independent Researcher
[4]Key Laboratory of Big Data Storage and Management, Northwestern Polytechnical University, Ministry of Industry and Information Technology, Xi'an
lysong@nwpu.edu.cn, xiaolinh@nwpu.edu.cn
panyudai@nwpu.edu.cn, shang@nwpu.edu.cn

Abstract

Multimodal Large Language Models (MLLMs) require continuous factual updates to stay current, yet lifelong knowledge editing remains a daunting challenge. The primary obstacle is the severe Locality Erosion and Catastrophic Forgetting, where sequential edits inevitably interfere with pre-trained capabilities and prior updates. In this paper, we propose **ROSE**, a **R**isk-aware **O**rthogonal **S**ubspace navigation framework for lifelong knowledge **E**diting. ROSE addresses forgetting by aiming to restrict parameter updates to the orthogonal complement of previously edited subspaces. Crucially, it introduces a risk-guided mechanism that utilizes a dynamically computed risk mask to safeguard specific factual parameter subspaces essential for model stability. During inference, ROSE employs a multi-granularity knowledge integration strategy, featuring prototype-anchored semantic gating and perturbation-driven subspace fusion to precisely activate edited knowledge for relevant queries while limiting interference on unrelated inputs. Comprehensive experiments on two prominent multimodal benchmarks show that ROSE is competitive with state-of-the-art approaches across five key metrics, while maintaining outstanding stability over $1,000$ sequential updates. Our code is available at https://github.com/sunlitsong/ROSE.

1 Introduction

Multimodal Large Language Models (MLLMs) have demonstrated remarkable capabilities in advanced reasoning and cross-modal understanding, largely due to their ability to align visual and textual semantics [1] and fuse information from diverse data sources [2–5]. However, these models remain tethered to the static knowledge captured during their pre-training on fixed datasets. This "knowledge freezing" limits their ability to adapt to a rapidly evolving world, frequently resulting in hallucinated content and outdated responses when faced with new events [6, 7]. While conventional full-parameter fine-tuning is a natural consideration for updates, it is computationally prohibitive and prone to catastrophic forgetting [8, 9]. Furthermore, most existing model editing methods are designed for static, isolated updates [10–12], making them ill-suited for the continuous, sequential data streams encountered in real-world scenarios.

*Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Consequently, the capacity for Lifelong Knowledge Editing (LKE) has become essential, requiring not only efficient modification of specific facts but also long-term reliability across a prolonged stream of knowledge [13–15]. In the multimodal domain, this task faces two challenges: (1) **Catastrophic Forgetting**, where new updates disrupt previously learned edits, and (2) **Locality Erosion**, where the model's performance on unrelated knowledge deteriorates. These issues are exacerbated by the complex modal entanglement between visual and textual representations. Such entanglement makes the model's knowledge space particularly sensitive to sequential perturbations, where a single factual update can inadvertently trigger a chain reaction of degradation across different modalities [16, 7].

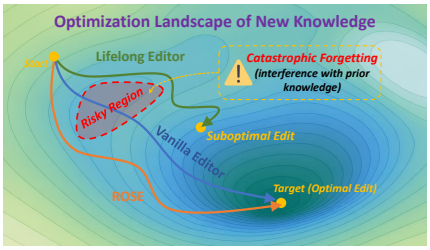

Figure 1: Comparison of optimization trajectories generated by conventional Editors and our ROSE. By navigating parameter updates away from high-risk general subspaces, ROSE achieves a superior balance between incorporating new knowledge and preserving pre-trained stability.

The fundamental issue lies in the unresolved tension between editing effectiveness and knowledge preservation. As illustrated in Figure 1, existing lifelong editors struggle to navigate this trade-off: they are often either overly conservative, which reduces the reliability of new edits, or overly intrusive, leading to knowledge interference that corrupts both pre-trained capabilities and prior edits [17, 18]. Crucially, these approaches lack subspace awareness, as they provide no guidance for pinpointing which parameter subspaces are susceptible to degradation. Consequently, as edits accumulate, the boundaries of modified knowledge become increasingly blurred, leading to "false triggers" where edited knowledge is unintentionally activated for irrelevant inputs, thereby undermining model locality [19, 13].

In this paper, we introduce ROSE, a new framework for Risk-aware Orthogonal Subspace navigation designed for lifelong knowledge Editing in MLLMs. ROSE operates on a "navigate-then-integrate" philosophy to safeguard the model's foundational capabilities during continuous updates. During the editing phase, it navigates weight updates into the orthogonal complement of previously stored knowledge, while employing a risk mask to anchor parameter subspaces essential for general stability. During the inference phase, it integrates the corresponding knowledge through a multi-granularity knowledge integration strategy, using prototype similarity [20] to assess input relevance. By combining prototype-anchored gating with perturbation-driven fusion, ROSE selectively activates edited knowledge for pertinent inputs while routing queries rejected by the gate to the stable pre-trained backbone. This dual-phase approach ensures both high plasticity for new facts and robust reliability for existing capabilities. Our key contributions can be outlined as follows:

- **Risk-Aware Orthogonal Editing Framework**: We introduce ROSE, a novel framework for lifelong MLLM editing. It features a risk-guided orthogonal editing mechanism that aims to restrict parameter updates to the orthogonal complement of the existing knowledge subspace. This is augmented by a dynamic risk mask that anchors parameters critical for general pre-trained capabilities, effectively mitigating catastrophic forgetting.
- **Multi-Granularity Knowledge Integration**: We propose a coarse-to-fine integration strategy to bridge the gap between pre-trained and edited knowledge. By synergizing prototype-anchored semantic gating with perturbation-driven subspace fusion, this module adaptively routes and integrates representations during inference, significantly mitigating locality erosion by preventing unintended factual triggers.
- **Extensive Empirical Validation**: We conduct extensive evaluations on two prominent multimodal benchmarks. The results across five key metrics demonstrate that ROSE achieves state-of-the-art performance.

2 Background and Related works

2.1 Multimodal Large Language Models

Multimodal Large Language Models (MLLMs) have achieved remarkable strides in visual perception and linguistic understanding, including knowledge-based visual question answering [21] and image

captioning [3, 5, 4, 22]. To integrate complementary information across modalities [23], these models typically adopt a modular architecture comprising a vision encoder [24, 25], a vision-to-language projection module [22, 26], and a Large Language Model (LLM) backbone [27, 28]. Despite their prowess, MLLMs rely on static training datasets, causing their internal knowledge to become rapidly outdated. This limitation often manifests as hallucinations or incorrect responses to newly emerging facts. Crucially, the complex interplay between modalities [29] suggests that factual knowledge is not stored in isolation, but rather resides within specific parametric subspaces, a characteristic that motivates our subspace-based navigation approach.

2.2 Knowledge Editing

Knowledge editing seeks to modify specific factual associations within a model efficiently without retraining from scratch. Current paradigms can be categorized into four groups: (1) Locate-and-Edit techniques, such as ROME [10] and MEMIT [12], which pinpoint and update neurons encoding specific facts; (2) Meta-Learning approaches like MEND [11], which employ hypernetworks to predict weight updates from edit gradients; (3) Parameter-Efficient methods [15, 30] that attach trainable adapters to a frozen backbone; and (4) Retrieval-based methods [19, 31] that steer model behavior via in-context learning and forgo any parameter changes. While effective for isolated updates, these paradigms often lack a strategy to mitigate the overlapping interference that accumulates across a sequence of edits. This leads to a progressive degradation of both new and old knowledge, a limitation that necessitates a more robust mechanism for parameter isolation in lifelong scenarios.

2.3 Lifelong Knowledge Editing

Lifelong knowledge editing seeks to continually revise models while alleviating sequential catastrophic forgetting. Existing research primarily explores parameter isolation and modular expansion. For instance, RECIPE [13] utilizes sentinel-retrieved prompts to disentangle edits, while LEMoE [17] employs Mixture-of-Experts (MoE) architectures to distribute knowledge across different experts. Although these strategies have shown promise, scaling them to MLLMs remains difficult due to complex cross-modal entanglement, which LiveEdit [18] attempts to address by employing distinct routing mechanisms. However, while routing facilitates knowledge localization, these modular approaches often overlook the internal optimization trajectory, failing to prevent subspace collisions during the update of shared parameters or protect factual subspaces critical for pre-trained knowledge. ROSE fills this gap by introducing risk-aware orthogonal navigation, providing a rigorous geometric and risk-based constraint to safeguard model integrity at the parameter level.

3 Preliminaries

In this section, we provide a formal definition of the multimodal lifelong knowledge editing task and its associated evaluation metrics, following the conventions established in [32] and [31].

3.1 Multimodal Lifelong Knowledge Editing

Let $f(v, p; \theta)$ denote a MLLM parameterized by θ, where $v \in \mathcal{V}$ is an input image and $p \in \mathcal{P}$ is a textual prompt. The model generates an output $o = f(v, p; \theta)$.

Knowledge Editing. The goal of knowledge editing is to modify the model's behavior on a specific factual instance $e = (v_e, p_e, o_e \rightarrow o_e^*)$, where o_e is the original (incorrect or outdated) response and o_e^* is the target correct output. An editor $\mathcal{E}(f, e)$ aims to produce an updated parameter set θ' such that $f(v_e, p_e; \theta') = o_e^*$.

Lifelong Setting. In the lifelong knowledge editing scenario, the model encounters a continuous stream of edits $\mathcal{D}_{\text{edit}} = \{e_1, e_2, \ldots, e_T\}$. At each time step t, the model updates its parameters from θ_{t-1} to θ_t using the current edit e_t. The fundamental difficulty lies in managing the sequential interference within the parameter space, where updates for e_t may catastrophically disrupt the representations learned for prior edits or erode the model's inherent pre-trained capabilities.

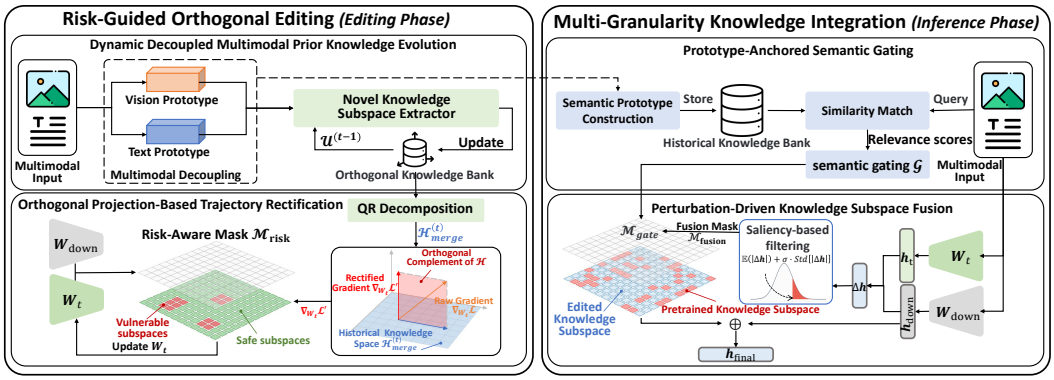

Figure 2: An illustration of the ROSE framework. It mainly consists of two core modules: the Risk-Guided Orthogonal Editing and the Multi-Granularity Knowledge Integration.

3.2 Evaluation Metrics

We evaluate ROSE across three core dimensions: (1) **Reliability** quantifies how effectively the model incorporates the specific target knowledge it was modified to handle; (2) **Generality** evaluates whether the edited knowledge generalizes to semantically equivalent samples within a neighborhood, assessed via rephrased texts (T-Gen.) and visually transformed images (M-Gen.); and (3) **Locality** safeguards the integrity of the model's original pre-trained knowledge against unintended interference on unrelated textual (T-Loc.) and multimodal (M-Loc.) samples. The formal mathematical definitions for these metrics are provided in the Appendix A.

4 Methodology

In this section, we introduce the ROSE framework, designed to alleviate two distinct types of catastrophic forgetting in lifelong editing: the degradation of previously applied edits and the erosion of the model's inherent pre-trained capabilities. As illustrated in Fig.2, ROSE consists of two synergistic modules: (1) a Risk-Guided Orthogonal Editing mechanism for safe knowledge absorption, and (2) a Multi-Granularity Knowledge Integration strategy for adaptive knowledge retrieval during inference. The complete procedural workflow is summarized in Appendix H.

4.1 Risk-Guided Orthogonal Editing Mechanism

Factual knowledge primarily resides in the MLP down-projection matrices. We therefore target $\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d_m \times d_n}$ at the l-th layer, where d_m and d_n represent the output and input dimensions, respectively. To preserve pre-trained capabilities, we freeze the backbone and isolate sequential updates within an auxiliary branch. Let $\boldsymbol{W}_t$ denote the parameters of this auxiliary branch at editing step t, initialized as $\boldsymbol{W}_0 = \boldsymbol{W}_{\text{down}}$ before any edits occur.

For the t-th edit $e_t = (v_t, p_t, o_t) \in \mathcal{D}_{\text{edit}}$, let $\boldsymbol{x}^{(t)} \in \mathbb{R}^{d_n \times L}$ denote its input features, where L denotes the token length. To prevent catastrophic forgetting of prior edits, the ideal weight update $\Delta \boldsymbol{W}_t$ aims to satisfy the following orthogonal constraint:

$$(\boldsymbol{W}_t + \Delta \boldsymbol{W}_t)\boldsymbol{x}_{\text{pre}} = \boldsymbol{h}_{\text{pre}}, \tag{1}$$

where $\boldsymbol{x}_{\text{pre}}$ and $\boldsymbol{h}_{\text{pre}}$ represent the input and output features of the edited samples, respectively. Given that $\boldsymbol{W}_t \cdot \boldsymbol{x}_{\text{pre}} = \boldsymbol{h}_{\text{pre}}$ already holds, this reduces to requiring $\Delta \boldsymbol{W}_t$ to lie in the null space of past inputs, i.e., $\Delta \boldsymbol{W}_t \cdot \boldsymbol{x}_{\text{pre}} = \boldsymbol{0}$.

Dynamic Decoupled Multimodal Prior Knowledge Evolution. To alleviate cross-modal entanglement, we decouple $\boldsymbol{x}^{(<t)}$ into a visual subspace $\mathcal{H}_v^{(<t)}$ and a textual subspace $\mathcal{H}_p^{(<t)}$. Specifically, we decompose each past input $\boldsymbol{x}^{(t-1)}$ into visual tokens $\boldsymbol{x}_v^{(t-1)}$ and textual query tokens $\boldsymbol{x}_p^{(t-1)}$. Then

we encode the visual modality by the centroid $\boldsymbol{x}'_v = \frac{1}{L_v}\sum_{i=1}^{L_v} \boldsymbol{v}_i$ and the textual modality by the final query token $\boldsymbol{x}'_p$. This aggregation balances semantic expressiveness and computational efficiency.

To effectively manage the lifelong accumulation of knowledge over time, we maintain an orthogonal matrix $\boldsymbol{U}_* \in \mathbb{R}^{d_n \times t}$ that spans the existing knowledge subspace $\mathcal{H}_*$. Here, $* \in \{v, p\}$. When a new edit feature $\boldsymbol{x}'_* \in \{\boldsymbol{x}'_p, \boldsymbol{x}'_v\}$ arrives, we first normalize it as $\hat{\boldsymbol{x}}_* = \boldsymbol{x}'_*/\|\boldsymbol{x}'_*\|_2$. To embed this new direction into the existing subspace while maintaining orthogonality, we project $\hat{\boldsymbol{x}}_*$ onto the orthogonal complement of $\boldsymbol{U}_*^{(t-1)}$, yielding a residual $\tilde{\boldsymbol{x}}_*$ that is independent of all preceding edits:

$$\tilde{\boldsymbol{x}}_* = \hat{\boldsymbol{x}}_* - \boldsymbol{U}_*^{(t-1)}(\boldsymbol{U}_*^{(t-1)})^\top \hat{\boldsymbol{x}}_*. \tag{2}$$

This process effectively filters out the information already captured by earlier edits, so that $\boldsymbol{U}_*$ becomes a matrix whose columns are the existing orthonormal basis vectors. To ensure numerical stability and keep the subspace compact, we only expand the basis when the L_2 norm of the residual is greater than a predefined threshold ϵ:

$$\boldsymbol{U}_*^{(t)} \leftarrow \begin{cases} \left[\boldsymbol{U}_*^{(t-1)}, \frac{\tilde{\boldsymbol{x}}_*}{\|\tilde{\boldsymbol{x}}_*\|_2}\right], & \text{if } \|\tilde{\boldsymbol{x}}_*\|_2 > \epsilon. \\ \boldsymbol{U}_*^{(t-1)}, & \text{otherwise.} \end{cases} \tag{3}$$

Orthogonal Projection-Based Trajectory Rectification. By iteratively updating $\boldsymbol{U}_*$ through this subspace-aware mechanism, we ensure it effectively captures the edited core knowledge space. To satisfy the constraint $\Delta \boldsymbol{W}_t \cdot \boldsymbol{x}_{\text{pre}} = \boldsymbol{0}$ established in Equation (1), the weight update must be restricted to the null space of $\mathcal{H}_*$. Because $\boldsymbol{U}_v$ and $\boldsymbol{U}_p$ arise from distinct modalities, we combine them into a single representation $\boldsymbol{U}_{\text{merged}}^{(t)} = [\boldsymbol{U}_v^{(t)}, \boldsymbol{U}_p^{(t)}]$. We then apply a QR decomposition to this merged matrix to obtain a strictly orthonormal basis for the resulting subspace:

$$\boldsymbol{Q}^{(t)}, \boldsymbol{R}^{(t)} = \text{QR}(\boldsymbol{U}_{\text{merged}}^{(t)}), \tag{4}$$

where $\boldsymbol{Q}^{(t)} \in \mathbb{R}^{d_n \times 2t}$ denotes an orthonormal basis of the prior knowledge subspace $\mathcal{H}_v^{(t)} \bigcup \mathcal{H}_p^{(t)}$.

While projecting gradients away from $\boldsymbol{Q}^{(t)}$ helps protect explicit prior edits, it leaves the vast pre-training distribution vulnerable. Inspired by Task Vectors [33], we quantify parameter vulnerability via absolute weight subtraction, as large deviations naturally highlight the specific parameters responsible for acquiring new facts. To preserve inherent general capabilities, we restrict updates to only the most active parameters. Compared to complex gradient-based metrics, this highly efficient strategy effectively pinpoints the factual parameter subspace using a risk mask $\boldsymbol{\mathcal{M}}_{\text{risk}}$:

$$\boldsymbol{\mathcal{M}}_{\text{risk}} = \left[\mathbb{I}\big(|\Delta \boldsymbol{W}_{i,j}| > \psi_k\big)\right]_{i,j}, \tag{5}$$

where $\Delta \boldsymbol{W} = \boldsymbol{W}_t - \boldsymbol{W}_{\text{down}}$, ψ_k is the $(1-k)$-th quantile of absolute differences $\{|\Delta \boldsymbol{W}_{i,j}|\}$, and $k \in (0, 1]$ dictates the selection fraction. Using the value captures weight plasticity, and employing $\boldsymbol{W}_{\text{down}}$ as a static anchor prevents cumulative error drift (further discussed in Appendix D).

Next, we integrate the orthogonal constraint with the empirically motivated risk mask to rectify the raw optimization gradient $\nabla_{\boldsymbol{W}_t}\mathcal{L}$:

$$\nabla_{\boldsymbol{W}_t}\mathcal{L} \leftarrow \left(\nabla_{\boldsymbol{W}_t}\mathcal{L} - (\nabla_{\boldsymbol{W}_t}\mathcal{L} \cdot \boldsymbol{Q}^{(t)})(\boldsymbol{Q}^{(t)})^\top\right) \odot \boldsymbol{\mathcal{M}}_{\text{risk}}, \tag{6}$$

where $\mathcal{L}$ is the loss function, and $\odot$ is the Hadamard product. Crucially, $\boldsymbol{\mathcal{M}}_{\text{risk}}$ is recomputed at each gradient step from the current $\boldsymbol{W}_t$. The inner projection steers gradients away from the prior knowledge subspace $\boldsymbol{Q}^{(t)}$, while $\boldsymbol{\mathcal{M}}_{\text{risk}}$ structurally regularizes the update by suppressing parameters critical for pre-trained capabilities.

4.2 Multi-Granularity Knowledge Integration Strategy

To mitigate erosion of pretrained knowledge, we propose a multi-granularity routing mechanism that dynamically determines the scope of each edit and integrates relevant knowledge spaces during inference, thereby maximizing the preservation of the model's locality.

Semantic Prototype Construction. To maintain a persistent record of the semantic scope for each update, we establish a Knowledge Bank $\mathcal{K}_*$. For each edit e, we derive its prototype $\boldsymbol{k}_*$ by aggregating

the representations of the corresponding modality tokens. Specifically, for an input $\boldsymbol{x} \in \mathbb{R}^{d_n \times L}$ during the editing phase, we update this bank by concatenating the normalized centroid:

$$\boldsymbol{k}_* = \frac{\sum_{i \in \mathcal{I}_*} \boldsymbol{x}_i}{\left\| \sum_{i \in \mathcal{I}_*} \boldsymbol{x}_i \right\|_2}, \quad \mathcal{K}_* \leftarrow [\mathcal{K}_*, \boldsymbol{k}_*], \tag{7}$$

where $\mathcal{I}_*$ denotes the set of token indices belonging to modality $* \in \{p, v\}$ and $\| \cdot \|_2$ denotes the L_2 normalization operator. This cumulative structure allows the model to preserve the individual semantic representation of each edit without the information compression inherent in orthogonalization.

At inference time, for any given input $\boldsymbol{x}$, we begin by assessing how relevant it is to the edited knowledge. To do this, we derive a global query vector $\boldsymbol{q}_*$ by computing the centroid specific to its modality: $\boldsymbol{q}_* = \sum_{i \in \mathcal{I}_*} \boldsymbol{x}_i / \left\| \sum_{i \in \mathcal{I}_*} \boldsymbol{x}_i \right\|_2$.

Prototype-Anchored Semantic Gating. Acting as a semantic router, this module directs inputs by quantifying their semantic affinity to prior knowledge subspaces $\mathcal{H}_p$ and $\mathcal{H}_v$. We compute the maximal projection alignment of the L_2-normalized query $\boldsymbol{q}_*$ onto the directional bases in the knowledge bank $\boldsymbol{\mathcal{K}}_*$. Formally, the relevance score s_* is defined as the maximum cosine similarity over the stored prototypes:

$$s_* = \max_{j \in \{1, \ldots, N\}} \langle \boldsymbol{q}_*, \boldsymbol{k}_{*,j} \rangle, \quad \text{where } * \in \{v, p\}. \tag{8}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, and N denotes the current number of prototypes stored in $\mathcal{K}_*$. It essentially captures the highest possible alignment between the input features and any single basis vector in the knowledge subspace, and guarantees that the router activates the auxiliary branch only when the input lies sufficiently close to a previously modified knowledge subspace.

The semantic gating $\mathcal{G} \in \{0, 1\}$ is then formulated by

$$\mathcal{G} = \mathbb{I}\Big(\bigwedge_{* \in \{v,p\}} \big(s_* > \tau_{\min}\big) \ \wedge \ \Big(\bigvee_{* \in \{v,p\}} \big(s_* > \tau\big) \ \vee \ |s_v - s_p| < \tau_{\text{gap}} \Big) \Big), \tag{9}$$

where $\tau_{\min}$ and τ_{gap} serve as safeguards that exclude only obviously irrelevant or cross-modally inconsistent inputs, leaving τ as the principal semantic threshold that governs the routing behavior. With this mechanism, the auxiliary branch is activated only when the input demonstrates strong linguistic or visual correspondence with the edited distribution.

Perturbation-Driven Knowledge Subspace Fusion. Since factual knowledge in MLLMs tends to be encoded in distinct subspaces, we capture the output discrepancy between the two branches as $\Delta \boldsymbol{h} = \boldsymbol{h}_t - \boldsymbol{h}_{\text{down}} \in \mathbb{R}^{d_m \times L}$, where $\boldsymbol{h}_t$ and $\boldsymbol{h}_{\text{down}}$ are the outputs of $\boldsymbol{W}_t$ and $\boldsymbol{W}_{\text{down}}$ for the same input. Under the linear logit mapping analyzed in Appendix B, the first-order sensitivity analysis in Corollary B.6 motivates using per-dimension perturbation magnitudes $|\Delta \boldsymbol{h}_{i,j}|$ as a tractable criterion for selective fusion. Accordingly, we construct the perturbation-driven fusion mask $\boldsymbol{\mathcal{M}}_{\text{fusion}} \in \{0, 1\}^{d_m \times L}$ by retaining only those knowledge subspaces whose perturbation exceeds a data-driven threshold:

$$\boldsymbol{\mathcal{M}}_{\text{fusion}} = \left[\mathbb{I}\left(|\Delta \boldsymbol{h}_{i,j}| > \mathbb{E}[|\Delta \boldsymbol{h}|] + \sigma \cdot \text{Std}[|\Delta \boldsymbol{h}|] \right) \right]_{i,j}, \tag{10}$$

where σ is a hyperparameter, and $\mathbb{E}$ and Std represent the expectation and standard deviation, respectively. The integrated output is regulated by a mask $\boldsymbol{\mathcal{M}}_{\text{gate}} = \mathcal{G} \cdot \boldsymbol{\mathcal{M}}_{\text{fusion}} \in \{0, 1\}^{d_m \times L}$, which in turn enables the integration of pretrained and edited knowledge:

$$\begin{aligned} \boldsymbol{h}_{\text{final}} &= \boldsymbol{h}_{\text{down}} + \boldsymbol{\mathcal{M}}_{\text{gate}} \odot \Delta \boldsymbol{h} \\ &= (1 - \boldsymbol{\mathcal{M}}_{\text{gate}}) \odot \boldsymbol{h}_{\text{down}} + \boldsymbol{\mathcal{M}}_{\text{gate}} \odot \boldsymbol{h}_t, \end{aligned} \tag{11}$$

where $\odot$ denotes the Hadamard product. This hierarchical design allows the model to selectively modify particular knowledge subspaces for the edited facts, while preserving the original pre-trained representations across most of the overall knowledge space.

5 Experiments

5.1 Experimental Setup

Benchmarks. Following [18], we assess ROSE on two widely used benchmarks: (1) MMEdit [16], the first multimodal editing benchmark that includes both VQA and image captioning; and (2)

Table 1: Performance comparison (%) on LLaVA-v1.5 (7B), with baseline results taken from Chen et al. [18]. Models are assessed on five primary aspects: Reliability (Rel.), Text Generality (T-Gen.), Multimodal Generality (M-Gen.), Text Locality (T-Loc.), and Multimodal Locality (M-Loc.) after 1, 100, and 1000 sequential edits.

Edits	Methods	VLKEB					E-IC					E-VQA				
		Reliability	Generality		Locality		Reliability	Generality		Locality		Reliability	Generality		Locality	
			T-Gen.	M-Gen.	T-Loc.	M-Loc.		T-Gen.	M-Gen.	T-Loc.	M-Loc.		T-Gen.	M-Gen.	T-Loc.	M-Loc.
T=1	FT-L	94.29	87.00	92.22	91.16	91.37	73.48	72.98	65.79	99.28	99.06	93.88	87.98	80.25	99.61	94.78
	FT-M	76.31	65.57	59.43	100.00	92.35	56.19	56.55	49.94	100.00	100.00	87.29	76.11	53.23	100.00	96.95
	MEND	92.13	91.28	90.22	89.19	90.13	92.82	91.81	90.59	96.38	93.69	91.23	90.05	91.29	91.02	90.22
	SERAC	89.77	89.11	87.92	66.68	14.20	88.18	81.03	85.61	84.01	28.58	89.33	83.72	84.97	82.05	23.78
	LEMoE	94.85	93.09	91.67	87.03	87.88	93.80	91.42	90.61	95.14	93.00	93.60	92.77	89.99	99.28	96.98
	RECIPE	92.67	92.35	91.01	89.67	82.85	84.45	76.97	81.57	96.53	96.37	91.37	86.51	87.73	94.27	88.88
	LiveEdit	96.43	95.22	**93.72**	100.00	**100.00**	82.16	81.01	78.27	100.00	**100.00**	94.28	**94.51**	88.01	100.00	100.00
	ROSE	**100.00**	**98.85**	88.87	**100.00**	98.72	**100.00**	**94.23**	**94.90**	**100.00**	99.55	**100.00**	93.92	**92.92**	**100.00**	**100.00**
T=100	FT-L	75.41	73.67	74.16	70.01	82.05	65.08	60.90	58.46	86.82	89.19	79.67	70.05	64.07	83.47	54.44
	FT-M	60.60	59.79	56.29	100.00	68.07	59.41	55.44	52.16	100.00	72.93	82.90	72.83	47.26	100.00	43.39
	MEND	0.56	0.58	0.66	0.18	0.07	56.83	56.89	53.07	87.63	84.64	2.22	2.20	2.21	0.21	0.62
	SERAC	72.25	62.43	70.68	53.73	13.69	53.35	53.70	49.41	48.04	17.25	88.08	81.53	82.48	62.13	12.90
	LEMoE	83.07	75.36	71.72	54.09	49.68	55.36	53.14	51.12	87.77	81.35	42.41	36.60	34.33	78.57	53.28
	RECIPE	73.97	63.73	72.69	86.2	82.59	53.23	53.75	49.36	87.64	95.52	89.86	83.32	84.82	87.37	85.08
	LiveEdit	94.56	90.65	89.56	100.00	**100.00**	80.81	78.77	63.52	100.00	100.00	93.54	**92.34**	**85.89**	100.00	99.31
	ROSE	**98.22**	**95.00**	**93.80**	**100.00**	95.31	**93.28**	**87.55**	**82.30**	**100.00**	**100.00**	**97.00**	85.22	83.97	**100.00**	**99.38**
T=1000	FT-L	68.14	66.38	66.98	65.61	75.35	59.78	54.99	54.17	65.37	78.96	71.39	59.83	57.41	55.55	48.99
	FT-M	53.41	48.80	43.16	100.00	57.03	49.21	47.75	43.81	100.00	35.14	69.57	56.34	44.07	100.00	41.47
	MEND	0.03	0.05	0.07	0.06	0.08	54.39	54.14	50.99	83.87	80.60	0.04	0.05	0.05	0.08	0.09
	SERAC	60.93	56.49	60.06	52.94	15.04	52.93	53.44	49.01	49.91	16.65	85.57	75.58	82.01	62.46	15.69
	LEMoE	67.97	61.07	58.16	48.48	44.06	34.50	31.38	28.14	82.09	75.88	30.80	25.75	24.32	71.45	46.23
	RECIPE	62.00	56.84	61.50	85.37	82.07	53.11	53.48	48.99	87.93	94.84	87.00	76.81	83.09	86.95	87.03
	LiveEdit	92.22	83.97	82.75	100.00	**100.00**	72.80	69.95	57.05	100.00	**99.79**	92.93	**90.16**	**84.30**	100.00	**96.43**
	ROSE	**95.24**	**92.74**	**84.53**	**100.00**	88.39	**90.65**	**82.52**	**72.22**	**100.00**	97.42	**97.59**	80.85	77.81	**100.00**	95.52

VLKEB [32], a comprehensive knowledge editing benchmark constructed with real-world images for VQA, aimed at providing a challenging and high-quality evaluation. We report the image-captioning and visual question-answering tasks of MMEdit as E-IC and E-VQA, respectively. Details of the experimental setup are presented in Appendix C.

Baselines. We select widely used and state-of-the-art approaches as baselines, spanning fine-tuning-based methods, meta-learning approaches [11], retrieval-augmented methods [19], lifelong editing frameworks [13, 17, 18], as well as several other advanced techniques. Our experiments are conducted on a range of MLLM backbones, including LLaVA-v1.5 (7B, 13B) [5], MiniGPT-4 [34], and Qwen2-VL [35]. Owing to space limitations, Section 5.2 mainly presents results on LLaVA-v1.5-7B, while the results for the remaining backbones are deferred to Section E.

5.2 Main Results

Table 1 presents the performance on VLKEB and MMEdit under the multimodal lifelong editing setup. From these results, three key observations can be made.

1. ROSE effectively mitigates catastrophic forgetting. As shown in Table 1, while most baselines experience a sharp decline in Reliability as the number of edits increases (e.g., FT-L drops to 68.14% on VLKEB at $T = 1000$), ROSE maintains a superior 95.24%. From $T = 1$ to $T = 1000$, ROSE's VLKEB Reliability decreases by 4.76 percentage points, compared with 26.15 for FT-L. At $T = 1000$, ROSE also exceeds LiveEdit in Reliability by 17.85 points on E-IC and 4.66 points on E-VQA, extending this advantage across both MMEdit tasks. This stability stems from our risk-guided orthogonal mechanism, which successfully navigates weight updates away from sensitive pre-trained subspaces while isolating new knowledge into orthogonal complements.

2. Superior comprehension and generalization. On VLKEB, ROSE exhibits robust performance in T-Gen and M-Gen, outperforming the strongest baseline, LiveEdit, by a significant margin at $T = 1000$ (e.g., 92.74% vs 83.97% in T-Gen). This is achieved via our Perturbation-Guided Knowledge Subspace Fusion, which extracts and integrates only the salient subspaces encoding the new knowledge while maintaining the backbone's fundamental representational structure. This selective modification helps apply edited facts across diverse paraphrases and visual contexts, even when the input query substantially deviates from the original editing sample, while limiting changes to the model's built-in general capabilities.

Table 2: Ablation study of ROSE on the VLKEB benchmark.

Settings	Reliability	Generality		Locality		Average
		T-Gen	M-Gen.	T-Loc.	M-Loc.	
Fine Tuning	79.16	76.75	75.94	14.54	7.44	50.77
w/o Multimodal Decoupling	91.05	88.14	85.63	100.00	40.54	81.07
w/o Knowledge Integration	93.15	90.81	90.05	57.38	15.83	69.44
w/o Semantic Gating	91.98	89.71	89.02	76.87	25.97	74.71
w/o Knowledge Fusion	96.28	93.90	83.30	100.00	84.37	91.57
w/o Orthogonal Editing	78.71	76.00	71.05	100.00	85.50	82.25
w/o Risk Awareness	90.07	87.31	78.09	100.00	88.14	88.72
ROSE	95.24	92.74	84.53	100.00	88.39	92.18

3. Outstanding locality preservation. Maintaining the integrity of unrelated knowledge is a critical failure point for many editors (e.g., SERAC's M-Loc drops to 15.04% on VLKEB). In contrast, ROSE achieves strong Locality (100% T-Loc and 88.39% M-Loc). This is attributed to our coarse-to-fine integration strategy: the semantic alignment gating encourages irrelevant queries to bypass the auxiliary branch, while the perturbation-guided fusion executes granular modifications for relevant inputs, thereby limiting unintended factual interference.

Furthermore, consistent gains across LLaVA, MiniGPT-4, and Qwen2-VL (see Appendix E) further validate that ROSE is architecture-agnostic.

5.3 Ablation Studies

We conduct extensive ablation studies on the VLKEB benchmark (Table 2) to dissect the contribution of each core component in ROSE.

Role of Risk-Aware Mask. The mask $\mathcal{M}_{\text{risk}}$ in Equation (5) is designed to protect pre-trained general knowledge by confining updates to active parameters of edited knowledge. As shown in Table 2, omitting this mask results in a significant drop in both Reliability and Generality. In the absence of $\mathcal{M}_{\text{risk}}$, ROSE degenerates to enforcing orthogonal constraints over the entire parameter space. This over-constraint severely shrinks the feasible solution space, making it harder to converge to an optimal edit for new tasks. Furthermore, the results indicate that without parameter isolation, edits inadvertently impair the model's foundational representations, confirming that $\mathcal{M}_{\text{risk}}$ is essential for maintaining the balance between plasticity and stability.

Effect of Orthogonal Lifelong Editing. The orthogonal projection of gradients onto the complement of the subspace $Q^{(t)}$ is fundamental to ROSE's ability to perform lifelong editing. In Table 2, ROSE achieves 95.24% Reliability versus 78.71% without orthogonal editing, and 88.39% M-Loc versus 40.54% without multimodal decoupling. Removing the orthogonal constraint leads the model to suffer from immediate catastrophic forgetting, with its performance on earlier edits degrading quickly as more edits are applied. Additionally, removing multimodal decoupling introduces severe cross-modal entanglement, where textual updates disrupt visual representations (and vice versa). Together, these results underscore the complementary roles of orthogonal editing and independent handling of textual and visual subspaces in effective multimodal lifelong editing.

Impact of Dual-Hierarchy Router. ROSE balances plasticity and stability through two-tier filtering during inference. (1) The coarse-grained gate $\mathcal{G}$ prevents the auxiliary branch from being mistakenly activated by irrelevant inputs, thereby preserving model Locality. (2) The fine-grained, perturbation-based mask $\mathcal{M}_{\text{fusion}}$ ensures that only the salient knowledge subspaces are injected into the backbone's pretrained knowledge during fusion. Table 2 shows that omitting either component leads to incorrect routing or "knowledge leakage," where edited facts interfere with general queries. This validates that a hierarchical routing mechanism is necessary to achieve high-precision knowledge retrieval in complex multimodal scenarios.

Table 3: Impact of the risk sensitivity threshold k.

Settings	Reliability	Generality		Locality		Average
		T-Gen.	M-Gen.	T-Loc.	M-Loc.	
0.1	71.42	63.63	63.53	100.00	89.86	77.69
0.3	91.02	79.28	77.23	100.00	89.44	87.39
0.5	95.24	92.74	84.53	100.00	88.39	92.18
0.7	95.23	91.63	82.10	100.00	88.59	91.51
0.9	96.55	92.88	83.38	100.00	88.27	92.22
1.0	90.07	87.31	78.09	100.00	88.14	88.72

Table 4: Impact of the perturbation-guided fusion threshold σ.

Edits	Settings	Reliability	Generality		Locality		Average
			T-Gen.	M-Gen	T-Loc.	M-Loc.	
T=1	1	100.00	99.10	88.87	100.00	97.89	97.17
	2	100.00	98.85	88.87	100.00	98.72	97.29
	3	100.00	98.15	88.87	100.00	99.26	97.26
T=100	1	98.00	93.77	89.55	100.00	94.91	95.25
	2	98.22	95.00	93.80	100.00	95.31	96.47
	3	98.01	92.40	89.08	100.00	97.23	95.34
T=1000	1	94.52	91.30	81.57	100.00	86.30	90.74
	2	95.24	92.74	84.53	100.00	88.39	92.18
	3	93.10	85.65	79.52	100.00	92.76	90.21

5.4 Further Studies

Impact of the risk sensitivity threshold k. The quantile k in $\mathcal{M}_{\text{risk}}$ serves as a threshold that controls how sensitively risks are detected. As shown in Table 3, a small k (e.g., 0.1) adopts a conservative strategy, targeting only those subspaces exhibiting the most pronounced perturbations, which severely limits the acquisition of new knowledge and drops Reliability. Conversely, an overly large k (e.g., 0.9) places ROSE in a highly sensitive regime, in which a broader range of parameter perturbations is labeled as risky. This enlarges the solution space but unintentionally encroaches upon general-purpose subspaces, weakening Locality. We found that $k = 0.5$ strikes a reasonable balance, enabling ROSE to accurately localize risky subspaces while limiting erosion of foundational capabilities.

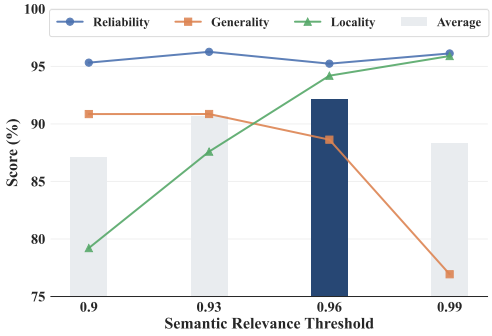

Figure 3: Sensitivity analysis of the semantic gating threshold τ. The "Average" represents the mean across all metrics.

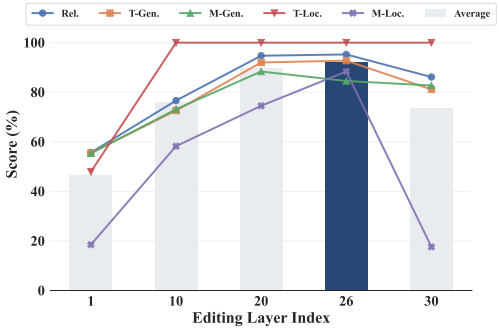

Figure 4: Effect of the attached layer index selected for editing. The "Average" denotes the mean value of all metrics.

Analysis of the Semantic Gating. We examine how the gating threshold τ affects performance (Figure 3), keeping $\tau_{\text{min}} = 0.4$ and $\tau_{\text{gap}} = 0.01$ fixed. While Reliability remains robust across a wide range of thresholds, Generality and Locality exhibit an intrinsically adversarial relationship. A low τ yields an overly permissive router that mistakenly triggers the editing branch for unrelated inputs, harming Locality. Conversely, a high τ improves Locality but struggles to recognize semantically rephrased queries, reducing Generality. The apparent sharp peak in the average metric reflects the balance between these two competing objectives. At $\tau = 0.96$, the router achieves a balance by accurately triggering edits for pertinent inputs while remaining safely inactive for the irrelevant distribution, thereby maximizing the model's overall stability.

Analysis of the saliency of Knowledge Fusion. The threshold σ controls the intensity of edited knowledge fusion (Table 4). A small σ leads to excessive mixing, injecting auxiliary feature into insignificant subspaces and causing interference. This unnecessary interference erodes pretrained knowledge and undermines locality. In contrast, a large σ causes under-mixing by confining fusion to only the strongest activation peaks. This suppresses more nuanced contextual dependencies, leading to incomplete semantics and reduced reliability. Setting $\sigma = 2$ yields highly targeted integration, capturing key novel semantics while maintaining the backbone's structure within invariant subspaces.

Effect of Editing Depth. We analyze how ROSE performs when integrated at different network depths. As shown in Figure 4, adding ROSE to early layers (e.g., 1-10) harms performance because

these layers primarily encode low-level perceptual details rather than abstract factual associations. Similarly, final layers contain highly compressed, task-specific representations that lack the semantic breadth needed for editing. The best results emerge when ROSE is applied to mid-to-late layers (e.g., Layer 26), indicating that factual knowledge is concentrated in an intermediate semantic space where strong cross-modal alignment supports accurate editing.

Effect of the Reference Anchor. The comparison in Appendix D shows better Locality preservation with the static backbone anchor $\boldsymbol{W}_{\text{down}}$ than with the sequential anchor $\boldsymbol{W}_{t-1}$. A fixed reference keeps risk identification tied to the original pre-trained weights throughout the editing sequence, supporting ROSE's use of a consistent reference for protecting foundational knowledge.

5.5 Analysis of Computational Cost

In lifelong knowledge editing over continuous and long data streams, computational efficiency is paramount. To evaluate the operational overhead of the multiple components within our framework (such as orthogonal projection and multi-granularity routing), we measured the average editing time, inference time, and peak memory consumption of ROSE. All efficiency evaluations were conducted on a single NVIDIA RTX 4090 GPU using the LLaVA-v1.5-7B backbone. The results are summarized in Table 5.

Table 5: Computational cost of ROSE across different numbers of sequential edits. Metrics include average editing time per edit, average inference time per query, and peak VRAM consumption.

Edits	Editing Time (s/edit)	Inference Time (s/query)	Peak VRAM (MB)
1	1.6748	0.9973	16781.52
10	1.7113	1.1849	16826.89
100	1.3975	1.1248	16857.73
1000	1.3140	1.1256	17005.69

As shown in Table 5, the **Editing Time** remains highly stable at approximately 1.3 to 1.7 seconds per edit, even as the sequence scales up to $1,000$ edits. This indicates that the orthogonal projection mechanism does not incur escalating computational burdens over time. Similarly, the **Inference Time** is highly consistent at around 1.1 seconds per query. This demonstrates that the proposed multi-granularity semantic gating and perturbation-driven fusion mechanisms introduce virtually no noticeable latency during the inference phase.

Furthermore, the **Peak VRAM** usage grows extremely slowly. Scaling from a single edit to $1,000$ continuous edits adds only about 224 MB of memory overhead. This confirms that our explicit parameter isolation strategy, utilizing auxiliary branches and maintaining compact orthogonal matrices, is highly memory-efficient, making ROSE practically viable for long-term lifelong editing scenarios.

For extreme lifelong scenarios where the accumulated feature subspace approaches full rank, we discuss a possible Singular Value Decomposition (SVD)-based basis truncation strategy (see Appendix F) to manage knowledge decay gracefully.

6 Conclusion

In this paper, we present ROSE, a lifelong knowledge editing framework for MLLMs that strikes a balance between plasticity and stability. By aiming to restrict parameter updates to the orthogonal complement of the edited knowledge subspace and applying a risk mask, ROSE navigates the knowledge space and reduces catastrophic interference between sequential edits. Moreover, our multi-granularity integration scheme enables precise activation of the relevant edited knowledge while limiting locality erosion. Extensive experiments demonstrate that ROSE consistently outperforms existing baselines.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (62576283, 62402390, 62607058, 62672417, 62302397), Natural Science Basic Research Program of Shaanxi (2026JC-YXQN-252, 2026JC-YXQN-234, 2026JC-YBQN-0841), Guangdong Basic and Applied Basic Research Foundation under Grant No. 2025A1515012995, Young Talent Fund of Xi'an Association for Science and Technology (0959202613046), Young Elite Scientists Sponsorship Program by CAST (YESS20240039), Postdoctoral Research Project Funding of Shaanxi Province (2025BSHTDZZ005, 2025BSHSDZZ106), China Postdoctoral Science Foundation (2025M774364), and Open Project Program of the State Key Laboratory of Medical Proteomics (SKLP-O202514).

References

[1] Lingyun Song, Xuequn Shang, Chen Yang, and Mingxuan Sun. Attribute-guided multiple instance hashing network for cross-modal zero-shot hashing. *IEEE Transactions on Multimedia*, 25:5305–5318, 2023. doi: 10.1109/TMM.2022.3190222.

[2] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

[3] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[4] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[5] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[6] Song Wang, Yaochen Zhu, Haochen Liu, Zaiyi Zheng, Chen Chen, and Jundong Li. Knowledge editing for large language models: A survey. *ACM Computing Surveys*, 57(3):1–37, 2024.

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

[8] Junhao Zheng, Shengjie Qiu, Chengming Shi, and Qianli Ma. Towards lifelong learning of large language models: A survey. *ACM Computing Surveys*, 57(8):1–35, 2025.

[9] Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. Investigating the catastrophic forgetting in multimodal large language models. *arXiv preprint arXiv:2309.10313*, 2023.

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

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

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

[13] Qizhou Chen, Taolin Zhang, Xiaofeng He, Dongyang Li, Chengyu Wang, Longtao Huang, et al. Lifelong knowledge editing for llms with retrieval-augmented continuous prompt learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13565–13580, 2024.

[14] Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *Advances in Neural Information Processing Systems*, 37:53764–53797, 2024.

[15] Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with grace: Lifelong model editing with discrete key-value adaptors. *Advances in Neural Information Processing Systems*, 36:47934–47959, 2023.

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

[17] Renzhi Wang and Piji Li. Lemoe: Advanced mixture of experts adaptor for lifelong model editing of large language models. *arXiv preprint arXiv:2406.20030*, 2024.

[18] Qizhou Chen, Chengyu Wang, Dakan Wang, Taolin Zhang, Wangyue Li, and Xiaofeng He. Lifelong knowledge editing for vision language models with low-rank mixture-of-experts. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9455–9466, 2025.

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

[20] Xiaolin Han, Xiurui Hu, Chenhao Ma, and Xuequn Shang. Explainable anomaly detection in dynamic heterogeneous graphs via relation evolution. *IEEE Transactions on Knowledge and Data Engineering*, 38 (5):2793–2806, 2026. doi: 10.1109/TKDE.2026.3665892.

[21] Lingyun Song, Jianao Li, Jun Liu, Yang Yang, Xuequn Shang, and Mingxuan Sun. Answering knowledge-based visual questions via the exploration of question purpose. *Pattern Recognition*, 133: 109015, 2023. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2022.109015. URL `https://www.sciencedirect.com/science/article/pii/S0031320322004952`.

[22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[23] Han-Wu Zhu, Hai-Cheng Yi, Zhu-Hong You, Shi-Hui He, and Xuequn Shang. A multimodal information fusion framework for accurate and robust prediction of drug–drug interactions. *Information Fusion*, 128:103981, 2026. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2025.103981. URL `https://www.sciencedirect.com/science/article/pii/S1566253525010437`.

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[25] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[26] Hezheng Lin, Xing Cheng, Xiangyu Wu, and Dong Shen. Cat: Cross attention in vision transformer. In *2022 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2022.

[27] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[28] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

[29] Lingyun Song, Wenqing Du, Xiaolin Han, Xinbiao Gan, Xiaoqi Wang, and Xuequn Shang. Cross-modal multi-relational graph reasoning: A novel model for multimodal textbook comprehension. *Information Fusion*, 120:103082, 2025. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2025.103082. URL `https://www.sciencedirect.com/science/article/pii/S1566253525001551`.

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

[31] Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? *arXiv preprint arXiv:2305.12740*, 2023.

[32] Han Huang, Haitian Zhong, Tao Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Vlkeb: A large vision-language model knowledge editing benchmark. *Advances in Neural Information Processing Systems*, 37:9257–9280, 2024.

[33] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

[34] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

[35] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[36] Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.

A Detailed Evaluation Metrics

In this appendix, we provide the formal definitions of the evaluation metrics used in knowledge editing: reliability, generality, and locality.

Reliability. Reliability quantifies the success rate of the edited model on the exact instances it was modified to handle. For a given edit $e = (v_e, p_e, o_e^*)$, where o_e^* represents the target output, reliability is defined as:

$$\text{Reliability} = \mathbb{E}_{e \sim \mathcal{D}_{\text{edit}}} \mathbb{I}\left\{ f(v_e, p_e; \theta') = o_e^* \right\}, \tag{12}$$

where $\mathbb{I}\{\cdot\}$ denotes the indicator function, and θ' represents the parameters of the edited model.

Generality. Generality measures the model's ability to apply the edited knowledge to semantically equivalent samples within a neighborhood $\mathcal{N}(e)$. These samples typically include rephrased textual prompts p_{re} or visually transformed images v_{re} that convey the same underlying fact as the original edit. The metric is formulated as:

$$\text{Generality} = \mathbb{E}_{e \sim \mathcal{D}_{\text{edit}}} \mathbb{E}_{(v_{\text{re}}, p_{\text{re}}) \sim \mathcal{N}(e)} \mathbb{I}\left\{ f(v_{\text{re}}, p_{\text{re}}; \theta') = o_e^* \right\}. \tag{13}$$

Locality. Locality evaluates whether the editing process unintentionally alters the model's behavior on unrelated knowledge. Let $\mathcal{D}_{\text{loc}}$ denote a set of irrelevant samples $(v_{\text{loc}}, p_{\text{loc}})$ that are independent of the edit e. Locality quantifies the extent to which the original pretrained knowledge is preserved:

$$\text{Locality} = \mathbb{E}_{(v_{\text{loc}}, p_{\text{loc}}) \sim \mathcal{D}_{\text{loc}}} \mathbb{I}\left\{ f(v_{\text{loc}}, p_{\text{loc}}; \theta') = f(v_{\text{loc}}, p_{\text{loc}}; \theta) \right\}, \tag{14}$$

where θ represents the parameters of the original pre-trained model.

B Mathematical Derivations and Proofs

In this appendix, we establish the formal mathematical foundation for analyzing representation-level sensitivity. This analysis directly inspires the perturbation-driven fusion mask utilized in our multi-granularity knowledge integration strategy.

Recent evidence [36] links distributional shifts to catastrophic forgetting during fine-tuning. Building on this insight, Theorem B.5 derives a first-order approximation of the fixed-target KL loss change caused by an intermediate perturbation under the linear mapping defined below, and Corollary B.6 bounds this change by a first-order contribution and a controlled remainder, thereby providing theoretical motivation for the perturbation-driven fusion mask in Section 4.2, where $|\Delta \boldsymbol{h}_{i,j}|$ serves as a computationally feasible surrogate for identifying actively modified feature coordinates. We first introduce the necessary definitions and lemmas, then present the proof of the theorem and its corollary.

B.1 Preliminaries

Definition B.1 (Logit Mapping and Probability)**.** To theoretically analyze representation-level sensitivity at an intermediate layer l, we simulate mapping its output representation $\boldsymbol{h} \in \mathbb{R}^{d_m}$ for a single token directly to the vocabulary space. Specifically, we obtain the logits $\boldsymbol{z} \in \mathbb{R}^D$ via a linear projection $\boldsymbol{z} = \boldsymbol{W}_l^\top \boldsymbol{h}$, where $\mathbf{W}_l \in \mathbb{R}^{d_m \times D}$. The predictive distribution $\boldsymbol{q} \in \mathbb{R}^D$ is then given by the softmax function:

$$\boldsymbol{q}_k(\mathbf{z}) = \frac{\exp(\boldsymbol{z}_k)}{\sum_{j=1}^{D} \exp(\boldsymbol{z}_j)}, \quad \forall k \in \{1, \ldots, D\}. \tag{15}$$

Definition B.2 (Objective Function)**.** Let $\boldsymbol{p} \in \mathbb{R}^D$ be the fixed target distribution, with $p_k \geq 0$ and $\sum_k p_k = 1$. We use the convention $0 \log 0 = 0$ and minimize the Kullback-Leibler divergence $\mathbb{D}_{\text{KL}}(\boldsymbol{p} \| \boldsymbol{q})$:

$$\mathcal{L}(\mathbf{z}) = \mathbb{D}_{\text{KL}}(\boldsymbol{p} \| \boldsymbol{q}) = -\sum_{k=1}^{D} \boldsymbol{p}_k \log \boldsymbol{q}_k(\boldsymbol{z}) + \sum_{k=1}^{D} \boldsymbol{p}_k \log \boldsymbol{p}_k. \tag{16}$$

To analyze the sensitivity of the objective, we first require the Jacobian of the softmax function and the gradient of the loss with respect to the logits.

Lemma B.3 (Jacobian of Softmax)**.** *The partial derivative of the softmax output q_k with respect to the logit z_i is:*

$$\frac{\partial q_k}{\partial z_i} = q_k(\delta_{ki} - q_i), \tag{17}$$

where δ_{ki} is the Kronecker delta, which is 1 if $k = i$ and 0 otherwise.

Proof. Applying the quotient rule to q_k:

$$\begin{aligned}
\frac{\partial q_k}{\partial z_i} &= \frac{\partial}{\partial z_i}\left(\frac{e^{z_k}}{\sum_j e^{z_j}}\right) \\
&= \frac{\frac{\partial e^{z_k}}{\partial z_i}(\sum_j e^{z_j}) - e^{z_k}\frac{\partial(\sum_j e^{z_j})}{\partial z_i}}{(\sum_j e^{z_j})^2} \\
&= \frac{e^{z_k}\delta_{ki}(\sum_j e^{z_j}) - e^{z_k}e^{z_i}}{(\sum_j e^{z_j})^2} \\
&= \frac{e^{z_k}}{\sum_j e^{z_j}}\delta_{ki} - \frac{e^{z_k}}{\sum_j e^{z_j}}\frac{e^{z_i}}{\sum_j e^{z_j}} \\
&= q_k\delta_{ki} - q_k q_i \\
&= q_k(\delta_{ki} - q_i).
\end{aligned}$$

□

Lemma B.4 (Gradient of KL Divergence)**.** *The gradient of the objective function $\mathcal{L}$ with respect to the logit vector $\mathbf{z}$ is:*

$$\nabla_{\mathbf{z}}\mathcal{L} = \boldsymbol{q} - \boldsymbol{p}. \tag{18}$$

Proof. Using the chain rule and Lemma B.3:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial z_i} &= -\sum_{k=1}^{D} p_k \frac{\partial \log q_k}{\partial z_i} = -\sum_{k=1}^{D} \frac{p_k}{q_k}\frac{\partial q_k}{\partial z_i} \\
&= -\sum_{k=1}^{D} \frac{p_k}{q_k} q_k(\delta_{ki} - q_i) \\
&= -\sum_{k=1}^{D} p_k(\delta_{ki} - q_i) \\
&= -\left(\sum_{k=1}^{D} p_k\delta_{ki} - \sum_{k=1}^{D} p_k q_i\right) \\
&= -(p_i - q_i\sum_{k=1}^{D} p_k).
\end{aligned}$$

Since $\sum_{k=1}^{D} p_k = 1$, we have $\frac{\partial \mathcal{L}}{\partial z_i} = q_i - p_i$. Thus, $\nabla_{\mathbf{z}}\mathcal{L} = \boldsymbol{q} - \boldsymbol{p}$. □

With these preliminary results in place, we now formally state and prove the main sensitivity theorem.

B.2 The Sensitivity of Knowledge Perturbation

Theorem B.5 (The Sensitivity of Knowledge Perturbation)**.** *Consider an intermediate representation $\boldsymbol{h} \in \mathbb{R}^{d_m}$ at layer l. Let $\boldsymbol{W}_l \in \mathbb{R}^{d_m \times D}$ map it to logits $\boldsymbol{z} \in \mathbb{R}^D$ via $\boldsymbol{z} = \boldsymbol{W}_l^\top \boldsymbol{h}$. With the KL divergence $\mathbb{D}_{KL}$, a target distribution $\boldsymbol{p} \in \mathbb{R}^D$, and a predictive distribution $\boldsymbol{q} = \sigma(\boldsymbol{z}) \in \mathbb{R}^D$ obtained via the softmax function σ, the sensitivity of $\mathbb{D}_{KL}(\boldsymbol{p}\|\boldsymbol{q})$ to a small intermediate perturbation $\boldsymbol{\delta}_l \in \mathbb{R}^{d_m}$ at layer l can be written as:*

$$\Delta\mathbb{D}_{KL}(\boldsymbol{\delta}_l) = (\boldsymbol{q} - \boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l + \mathcal{O}(\|\boldsymbol{\delta}_l\|^2), \tag{19}$$

where $\Delta\mathbb{D}_{KL}(\boldsymbol{\delta}_l) = \mathcal{L}(\boldsymbol{z} + \boldsymbol{W}_l^\top \boldsymbol{\delta}_l) - \mathcal{L}(\boldsymbol{z})$, with $\boldsymbol{p}$ and $\boldsymbol{W}_l$ held fixed, and $\mathcal{O}(\|\boldsymbol{\delta}_l\|_2^2)$ encapsulates the higher-order terms.

Proof. Let $\mathbf{h} \in \mathbb{R}^{d_m}$ denote the latent representation at layer l, and let $\boldsymbol{\delta}_l \in \mathbb{R}^{d_m}$ be a small additive perturbation. The mapping from the latent space to the logits $\mathbf{z} \in \mathbb{R}^D$ is defined by a linear transformation $\mathbf{W}_l \in \mathbb{R}^{d_m \times D}$, such that $\mathbf{z} = \mathbf{W}_l^\top \mathbf{h}$. Consequently, the perturbation induced in the logit space, denoted as $\mathbf{v} \in \mathbb{R}^D$, is given by:

$$\mathbf{v} = \Delta\mathbf{z} = \mathbf{W}_l^\top(\mathbf{h} + \boldsymbol{\delta}_l) - \mathbf{W}_l^\top \mathbf{h} = \mathbf{W}_l^\top \boldsymbol{\delta}_l. \tag{20}$$

The objective function is the KL divergence $\mathcal{L}(\mathbf{z}) = \mathbb{D}_{\mathrm{KL}}(\mathbf{p}\|\mathbf{q}(\mathbf{z}))$. Our goal is to analyze the sensitivity variation $\Delta\mathbb{D}_{\mathrm{KL}}(\boldsymbol{\delta}_l) = \mathcal{L}(\mathbf{z} + \mathbf{v}) - \mathcal{L}(\mathbf{z})$.

We construct a parameterized scalar function $g(t) = \mathcal{L}(\mathbf{z} + t\mathbf{v})$ for $t \in [0, 1]$. For $t \in (0, 1]$, Taylor's theorem with Lagrange remainder around $t = 0$ gives:

$$g(t) = g(0) + g'(0)t + \frac{1}{2}g''(\xi_t)t^2, \qquad 0 < \xi_t < t, \tag{21}$$

where $g''(s) = \mathbf{v}^\top H(\mathbf{z} + s\mathbf{v})\mathbf{v}$ and $H(\mathbf{z}) = \nabla^2\mathcal{L}(\mathbf{z}) = \mathrm{diag}(\boldsymbol{q}(\mathbf{z})) - \boldsymbol{q}(\mathbf{z})\boldsymbol{q}(\mathbf{z})^\top$. At $t = 0$, the equality holds directly.

According to Taylor's Theorem, the total variation is:

$$\Delta\mathbb{D}_{\mathrm{KL}}(\boldsymbol{\delta}_l) = g(1) - g(0) = g'(0) + R_1(\boldsymbol{\delta}_l), \tag{22}$$

where $R_1(\boldsymbol{\delta}_l) = \frac{1}{2}\mathbf{v}^\top H(\mathbf{z} + \xi\mathbf{v})\mathbf{v}$ for some $\xi \in (0, 1)$. The Hessian H is positive semidefinite because its quadratic form is a variance under $\boldsymbol{q}$. Its maximum absolute row sum is $\max_i 2q_i(1-q_i) \leq 1/2$; since H is symmetric, $\|H\|_2 \leq 1/2$. Hence,

$$0 \leq R_1(\boldsymbol{\delta}_l) \leq \frac{1}{4}\|\mathbf{v}\|_2^2 = \frac{1}{4}\|\mathbf{W}_l^\top \boldsymbol{\delta}_l\|_2^2. \tag{23}$$

To find the linear term $g'(0)$, we apply the multivariate Chain Rule:

$$g'(t) = \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{z} + t\mathbf{v})^\top \frac{\partial}{\partial t}(\mathbf{z} + t\mathbf{v}) = \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{z} + t\mathbf{v})^\top \mathbf{v}. \tag{24}$$

Evaluating at $t = 0$, we have $g'(0) = \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{z})^\top \mathbf{v}$.

By Lemma B.4, the gradient of the KL divergence with respect to the logits is given explicitly by $\nabla_{\mathbf{z}}\mathcal{L} = \mathbf{q} - \mathbf{p}$. Substituting this result into the expression for $g'(0)$, we obtain:

$$g'(0) = (\mathbf{q} - \mathbf{p})^\top \mathbf{v}. \tag{25}$$

Finally, we map the sensitivity back to the intermediate perturbation $\boldsymbol{\delta}_l$ by substituting $\mathbf{v} = \mathbf{W}_l^\top \boldsymbol{\delta}_l$:

$$\Delta\mathbb{D}_{\mathrm{KL}}(\boldsymbol{\delta}_l) = (\mathbf{q} - \mathbf{p})^\top(\mathbf{W}_l^\top \boldsymbol{\delta}_l) + R_1(\boldsymbol{\delta}_l). \tag{26}$$

For fixed $\mathbf{W}_l$, (23) gives $R_1(\boldsymbol{\delta}_l) = \mathcal{O}(\|\boldsymbol{\delta}_l\|_2^2)$, which completes the proof. □

Building on Theorem B.5, we now derive a worst-case upper bound that isolates the role of the feature perturbation magnitude, which directly motivates the perturbation-driven fusion mask in Section 4.2.

Corollary B.6 (Worst-Case Bound for KL Loss Change)**.** *During inference, the true target distribution $\boldsymbol{p}$ is unknown, making it impossible to directly compute the first-order sensitivity term $(\boldsymbol{q} - \boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l$ in Theorem B.5. However, we can circumvent this difficulty by deriving a conservative upper bound on the fixed-target KL loss change that isolates the role of the feature perturbation $\boldsymbol{\delta}_l$. This bound provides a theoretically grounded yet practically feasible motivation for identifying the most actively modified feature dimensions without requiring knowledge of $\boldsymbol{p}$.*

By applying Hölder's inequality to the first-order term in Theorem B.5 and retaining the remainder bound (23)*, we obtain:*

$$\begin{aligned} |\Delta\mathbb{D}_{KL}(\boldsymbol{\delta}_l)| &\leq \left|(\boldsymbol{q} - \boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l\right| + \frac{1}{4}\|\boldsymbol{W}_l^\top \boldsymbol{\delta}_l\|_2^2 \\ &\leq \|\boldsymbol{W}_l(\boldsymbol{q} - \boldsymbol{p})\|_1 \cdot \|\boldsymbol{\delta}_l\|_\infty + \frac{1}{4}\|\boldsymbol{W}_l^\top \boldsymbol{\delta}_l\|_2^2. \end{aligned} \tag{27}$$

Alternatively, an element-wise decomposition provides a finer-grained perspective on how individual feature dimensions contribute to the worst-case bound:

$$|\Delta\mathbb{D}_{KL}(\boldsymbol{\delta}_l)| \leq \sum_{i=1}^{d_m} \left|\left[\boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p})\right]_i\right| \cdot |(\boldsymbol{\delta}_l)_i| + \frac{1}{4}\|\boldsymbol{W}_l^\top \boldsymbol{\delta}_l\|_2^2. \tag{28}$$

Proof. We begin from the exact decomposition established in the proof of Theorem B.5:

$$\Delta\mathbb{D}_{\mathrm{KL}}(\boldsymbol{\delta}_l) = (\boldsymbol{q}-\boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l + R_1(\boldsymbol{\delta}_l). \tag{29}$$

Taking the absolute value and rewriting the inner product,

$$|\Delta\mathbb{D}_{\mathrm{KL}}(\boldsymbol{\delta}_l)| \leq \left|(\boldsymbol{q}-\boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l\right| + |R_1(\boldsymbol{\delta}_l)| = \left|\left\langle \boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p}),\, \boldsymbol{\delta}_l \right\rangle\right| + |R_1(\boldsymbol{\delta}_l)|, \tag{30}$$

where $\langle\cdot,\cdot\rangle$ denotes the standard Euclidean inner product and we have used the identity $(\boldsymbol{q}-\boldsymbol{p})^\top \boldsymbol{W}_l^\top \boldsymbol{\delta}_l = \langle \boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p}),\, \boldsymbol{\delta}_l\rangle$.

Applying the Hölder inequality in the form $|\langle \boldsymbol{u}, \boldsymbol{v}\rangle| \leq \|\boldsymbol{u}\|_1 \|\boldsymbol{v}\|_\infty$ with $\boldsymbol{u} = \boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p}) \in \mathbb{R}^{d_m}$ and $\boldsymbol{v} = \boldsymbol{\delta}_l \in \mathbb{R}^{d_m}$ yields the following bound on the first-order term, which together with (23) gives (27):

$$\left|\left\langle \boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p}),\, \boldsymbol{\delta}_l \right\rangle\right| \leq \|\boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p})\|_1 \cdot \|\boldsymbol{\delta}_l\|_\infty. \tag{31}$$

Similarly, expanding the first-order inner product and applying the triangle inequality gives:

$$\left|\left\langle \boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p}),\, \boldsymbol{\delta}_l \right\rangle\right| = \left|\sum_{i=1}^{d_m} \left[\boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p})\right]_i \cdot (\boldsymbol{\delta}_l)_i\right| \leq \sum_{i=1}^{d_m} \left|\left[\boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p})\right]_i\right| \cdot \left|(\boldsymbol{\delta}_l)_i\right|. \tag{32}$$

Adding the remainder bound (23) proves (28). Both bounds separate the unknown target mismatch $\boldsymbol{q}-\boldsymbol{p}$ from the observable feature perturbation $\boldsymbol{\delta}_l$ in the first-order term. Since $\boldsymbol{p}$ and $\boldsymbol{q}$ are probability vectors, we have $\|\boldsymbol{q}-\boldsymbol{p}\|_1 \leq 2$, and the gradient-related factor $\|\boldsymbol{W}_l(\boldsymbol{q}-\boldsymbol{p})\|_1$ is bounded by $2\|\boldsymbol{W}_l\|_{1\to 1}$, where $\|\boldsymbol{W}_l\|_{1\to 1} = \max_j \sum_i |(\boldsymbol{W}_l)_{ij}|$. Together with the quadratic remainder, this gives a worst-case bound on the fixed-target KL loss change under the linear mapping. For token j, we apply the analysis with $\boldsymbol{\delta}_l = \Delta\boldsymbol{h}_{:,j}$, whose coordinate magnitudes are $|\Delta\boldsymbol{h}_{i,j}|$. The quadratic remainder quantifies the approximation error, while the fusion rule uses these observed feature magnitudes as its practical selection criterion. This first-order sensitivity analysis motivates our perturbation-driven fusion strategy in Section 4.2: by constructing the fusion mask $\boldsymbol{\mathcal{M}}_{\text{fusion}}$ from the magnitude $|\Delta\boldsymbol{h}_{i,j}|$ of the output discrepancy, we use coordinate-wise feature modification as a computationally feasible criterion for selective integration. □

C Experimental Settings

In our experiments, ROSE is attached to the 26-th layer. Optimization is performed using the AdamW optimizer with a learning rate of 5×10^{-4}. Regarding the stopping criteria for an individual edit, the optimization loop continues until the loss drops below a threshold of 0.01 or a maximum of 25 steps is reached. The threshold k in $\boldsymbol{\mathcal{M}}_{\text{risk}}$ is set to 0.5, the fusion parameter $\sigma = 2$, and the expansion threshold $\epsilon = 1 \times 10^{-5}$.

D Impact of the Reference Anchor for Risk Identification

We examine how to choose the reference weight used to compute the gradient perturbation $\Delta\boldsymbol{W}$. In particular, we compare two strategies: the sequential anchor $\boldsymbol{W}_{t-1}$ and the *static anchor* $\boldsymbol{W}_{\text{down}}$ derived from the backbone. As illustrated in Figure 5, employing the sequential anchor results in lower multimodal Locality. We ascribe this behavior to cumulative errors: as the model undergoes continuous updates, the reference point $\boldsymbol{W}_{t-1}$ gradually drifts away from the original pre-trained knowledge space. As a result, computing sensitivity relative to this shifted base fails to identify the subspaces that are essential for maintaining the model's inherent capabilities. In contrast, using the static anchor $\boldsymbol{W}_{\text{down}}$ offers a stable baseline. This helps the risk mask $\mathcal{M}_{\text{risk}}$ reliably safeguard the original knowledge structure throughout sequential editing, thereby supporting our design choice.

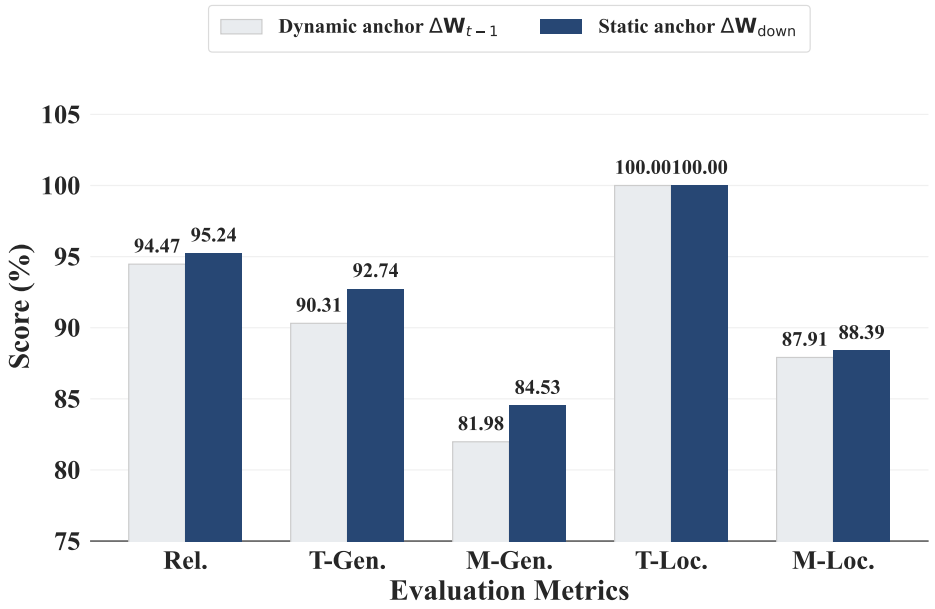

Figure 5: Impact of the Reference Anchor for Risk Identification.

E Impact of Different MLLM Backbones

To ensure the broad applicability and generalization of our approach, we extend the evaluation of ROSE beyond the default LLaVA-v1.5-7B model to encompass both different model architectures and a larger model scale. Specifically, we conduct comprehensive lifelong editing experiments on the MiniGPT-4 (7B) architecture, the Qwen2-VL (7B) model, and the larger LLaVA-v1.5-13B backbone.

First, we evaluate the performance of ROSE on the MiniGPT-4 (7B) model across the VLKEB, E-IC, and E-VQA benchmarks, as detailed in Table 6. The experimental results clearly demonstrate the robust adaptability of our framework to different network structures. Even after a long stream of sequential edits, ROSE maintains high reliability and exceptional locality across all three benchmarks. This confirms that the risk-aware orthogonal projection and multi-granularity knowledge integration mechanisms can effectively mitigate catastrophic forgetting and preserve the model's fundamental visual-linguistic capabilities, regardless of the specific architectural nuances of the MLLM.

Furthermore, to verify whether our mechanisms can effectively safeguard a significantly expanded parameter space and generalize to different model families, we scale our evaluation to the LLaVA-v1.5-13B backbone as well as the Qwen2-VL (7B) model. Figure 6 presents the performance comparison across these variants on the VLKEB benchmark after continuous edits. Notably, ROSE scales seamlessly to the larger 13B LLaVA model, yielding competitive results in the larger-model evaluation. Moreover, the results on the Qwen2-VL (7B) architecture further validate the versatility of our method across diverse model families, as it achieves highly competitive scores across all evaluation metrics and demonstrates strong robustness.

Overall, these extended evaluations indicate that the ROSE framework successfully mitigates catastrophic forgetting and maintains excellent performance across the evaluated model scales and architectures.

F Theoretical Discussions on Capacity Limits

A fundamental theoretical boundary exists in projection-based lifelong learning methods. When the accumulated historical feature subspace reaches full rank d_n, maintaining absolute orthogonality to all historical directions permits only the zero update. Under such conditions, acquiring novel facts through further weight updates requires a trade-off with protecting all historical feature directions. To

Table 6: Performance comparison (%) on MiniGPT-4 (7B), with baseline results taken from Chen et al. [18]. Models are assessed on five primary aspects: Reliability (Rel.), Text Generality (T-Gen.), Multimodal Generality (M-Gen.), Text Locality (T-Loc.), and Multimodal Locality (M-Loc.) after 1, 100, and 1000 sequential edits.

Edits	Methods	VLKEB					E-IC					E-VQA				
		Reliability	Generality		Locality		Reliability	Generality		Locality		Reliability	Generality		Locality	
			T-Gen.	M-Gen.	T-Loc.	M-Loc.		T-Gen.	M-Gen.	T-Loc.	M-Loc.		T-Gen.	M-Gen.	T-Loc.	M-Loc.
T=1	FT-L	82.17	81.64	78.45	98.45	75.06	60.40	59.47	52.79	99.95	92.46	93.85	86.25	88.58	99.47	84.50
	FT-M	93.51	91.28	68.30	100.00	81.78	71.60	67.34	49.55	100.00	91.69	91.42	85.82	49.09	100.00	86.72
	MEND	90.28	89.18	89.55	93.63	86.69	90.28	83.91	79.05	95.92	93.97	93.47	91.32	91.22	81.51	74.97
	SERAC	89.28	89.06	87.35	64.88	12.72	85.09	78.96	81.70	83.36	24.50	87.66	69.24	85.96	82.11	24.11
	TP	49.17	51.88	48.92	90.63	79.79	52.19	51.49	49.81	83.33	72.04	52.45	48.39	51.56	93.93	83.61
	LTE	92.92	85.16	86.25	88.29	81.96	91.30	84.62	83.55	94.96	91.88	93.54	77.59	84.07	94.03	88.14
	RECIPE	93.20	92.40	90.71	90.29	84.35	81.25	74.44	76.96	97.18	92.17	89.23	70.39	87.94	95.49	90.90
	LEMoE	92.26	89.28	88.27	93.87	67.49	90.32	86.30	85.75	96.66	69.17	92.59	90.49	89.16	97.01	84.33
	LiveEdit	96.41	95.56	**92.85**	100.00	**100.00**	86.84	83.82	76.95	100.00	**100.00**	94.90	92.63	**92.06**	100.00	**100.00**
	ROSE	**100.00**	**100.00**	87.38	**100.00**	97.81	**100.00**	**94.16**	**88.29**	**100.00**	96.79	**100.00**	**97.25**	87.67	**100.00**	99.27
T=100	FT-L	69.98	69.05	68.09	88.74	63.07	62.35	57.49	54.13	96.15	46.86	64.81	58.11	54.23	93.62	44.98
	FT-M	60.03	60.18	57.00	100.00	51.24	58.93	54.35	47.84	100.00	36.80	66.06	51.17	47.24	100.00	36.65
	MEND	38.85	40.02	37.99	72.82	68.89	51.92	53.96	48.24	86.02	83.88	23.32	20.75	21.42	47.27	42.60
	SERAC	69.38	58.31	67.57	51.71	13.72	47.41	46.17	45.22	49.37	13.52	87.39	66.53	81.70	65.13	14.04
	TP	39.60	42.52	39.14	36.15	43.68	39.60	39.64	38.48	17.29	26.04	19.46	20.32	22.03	41.85	32.91
	LTE	79.45	63.55	79.46	84.18	79.42	50.47	49.57	48.17	88.12	92.87	85.05	71.80	79.61	87.06	86.10
	RECIPE	71.54	60.04	70.35	85.57	86.85	47.27	46.06	45.12	89.52	94.25	88.26	67.74	81.48	91.77	88.23
	LEMoE	44.75	43.33	41.96	73.90	60.38	52.85	50.20	48.11	92.08	49.63	27.84	26.38	26.73	68.45	33.73
	LiveEdit	94.06	**94.62**	**89.42**	100.00	**100.00**	84.38	83.53	71.85	100.00	**100.00**	93.23	**91.27**	**88.34**	100.00	**99.61**
	ROSE	**95.42**	94.20	78.29	**100.00**	98.38	**91.41**	**85.41**	**81.30**	**100.00**	99.14	**94.97**	88.52	83.45	**100.00**	99.07
T=1000	FT-L	61.85	60.82	60.41	76.17	60.34	58.88	53.98	52.98	93.31	46.49	58.97	46.34	50.77	72.04	41.92
	FT-M	51.26	54.35	51.13	100.00	50.72	52.65	47.90	47.25	100.00	33.69	51.18	42.79	40.08	100.00	37.28
	MEND	42.35	45.07	42.68	66.98	62.32	49.30	51.46	46.70	81.03	82.53	31.84	26.60	33.98	43.08	44.90
	SERAC	57.02	51.66	56.80	48.48	13.24	47.03	45.79	44.82	42.12	15.31	84.50	60.36	81.83	63.26	13.40
	TP	23.84	25.00	23.54	18.79	25.58	25.07	24.50	24.09	14.77	18.05	9.24	8.96	10.25	20.54	17.03
	LTE	64.28	56.91	64.07	83.96	78.59	51.31	49.25	49.19	88.19	90.53	81.93	66.81	75.96	88.24	78.95
	RECIPE	58.18	52.31	58.14	85.49	83.84	47.02	45.76	44.85	89.80	92.26	85.33	61.79	76.35	90.33	89.08
	LEMoE	39.48	38.24	36.52	82.35	58.01	50.61	47.35	47.19	95.11	47.62	25.07	23.87	26.68	89.11	50.50
	LiveEdit	**93.41**	89.87	**85.21**	100.00	**100.00**	81.43	80.34	62.25	100.00	**99.73**	**92.75**	**89.81**	**85.12**	100.00	**96.82**
	ROSE	91.12	**89.92**	74.75	**100.00**	86.84	**86.90**	**80.76**	**70.93**	**100.00**	97.01	92.35	78.60	77.63	**100.00**	91.23

address this capacity limit, the ROSE framework can be naturally extended with a Singular Value Decomposition (SVD)-based compression of the historical feature matrix, whose columns contain historical feature vectors, retaining its leading left singular vectors as a reduced orthogonal basis.

According to the properties of SVD, discarding components associated with the smallest singular values yields an optimal low-rank approximation of the historical feature matrix in Frobenius or spectral norm. This form of controlled feature compression represents a mathematically sound compromise under strict finite-parameter constraints.

Our reported experiments evaluate sequences of up to $1,000$ sequential edits without capacity compression. Within this regime, the available update directions depend on the actual rank of the accumulated historical feature subspace. However, in extreme real-world deployments where continuous edits may scale to tens of thousands, this subspace may approach its dimensionality boundary. In such limiting scenarios, the aforementioned SVD-based truncation offers a possible extension for managing the finite subspace budget.

G Broader Impacts

Our research on lifelong knowledge editing for Multimodal Large Language Models (MLLMs) presents significant positive societal impacts. By enabling models to dynamically update their internal knowledge without catastrophic forgetting, ROSE effectively mitigates the issue of hallucinations and outdated factual responses. This can help ensure that deployed AI systems can remain factual, reliable, and aligned with the evolving real world, which is crucial for safety-critical applications such as healthcare, legal consultation, and education.

However, we acknowledge the potential negative societal impacts and dual-use risks associated with this technology. The ability to precisely and efficiently modify an MLLM's knowledge could be exploited by malicious actors to intentionally inject disinformation, bias, or harmful content into open-source models. Since ROSE is designed to seamlessly integrate edits while preserving the model's general capabilities, such poisoned models might behave normally on standard queries while outputting manipulative content for specific triggers. To mitigate these risks, we recommend that practitioners implement robust fact-checking verification pipelines before applying any edits.

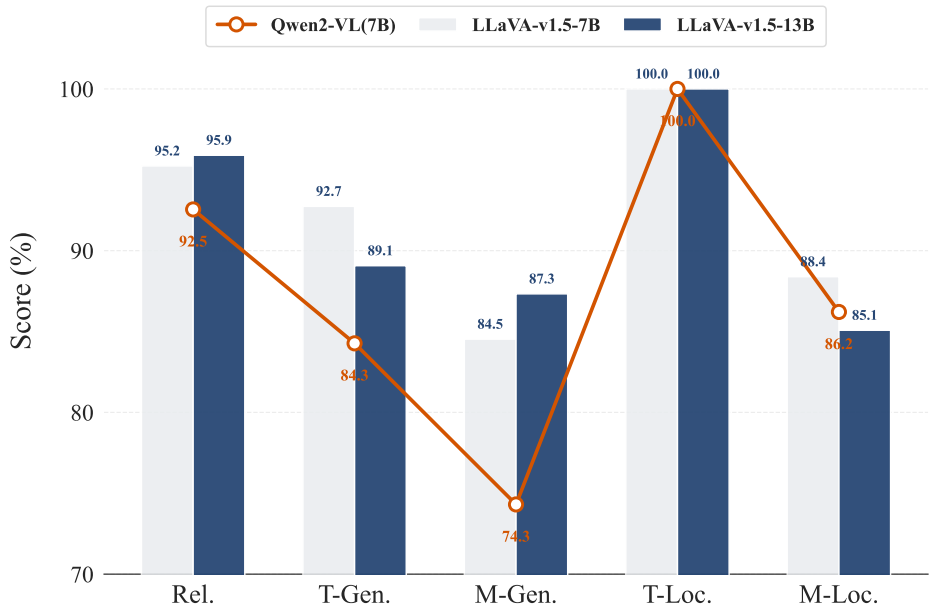

Figure 6: Performance comparison across LLaVA-v1.5-7B, LLaVA-v1.5-13B, and Qwen2-VL-7B backbones after 1,000 sequential edits on the VLKEB benchmark.

Furthermore, maintaining a transparent and auditable log of the Knowledge Bank is essential for tracking and reverting any unauthorized or malicious modifications in public-facing AI systems.

H Algorithmic Implementation Details of ROSE

To provide a clear and step-by-step overview of the ROSE framework, we summarize its complete execution procedure in Algorithm 1. The pseudocode concisely outlines our two primary stages: the lifelong editing phase for safely absorbing new sequential edits, and the inference phase for accurate, perturbation-guided knowledge routing.

Algorithm 1 ROSE: Risk-Aware Orthogonal Subspace Navigation for Lifelong Editing

Input: Pre-trained MLLM backbone with down-projection matrix $\boldsymbol{W}_{\text{down}}$; Sequential edit stream $\mathcal{D}_{\text{edit}} = \{e_1, e_2, \ldots, e_T\}$; Hyperparameters $k, \epsilon, \tau_{\min}, \tau, \tau_{\text{gap}}, \sigma$; Learning rate η.
Output: Updated auxiliary weights $\boldsymbol{W}_t$; Updated basis $\boldsymbol{U}^{(t)}$; Knowledge Bank $\boldsymbol{\mathcal{K}}$.

Initialize: $\boldsymbol{W}_0 \leftarrow \boldsymbol{W}_{\text{down}}$; $\boldsymbol{U}_v^{(0)}, \boldsymbol{U}_p^{(0)} \leftarrow \emptyset$; $\boldsymbol{Q}^{(0)} \leftarrow \emptyset$; $\boldsymbol{\mathcal{K}}_v, \boldsymbol{\mathcal{K}}_p \leftarrow \emptyset$.

#=== Phase 1: Lifelong Editing Phase (Risk-Guided Orthogonal Editing) ===#

for each edit $e_t = (v_t, p_t, o_t) \in \mathcal{D}_{\text{edit}}$ **do**
  Initialize current weights $\boldsymbol{W}_t \leftarrow \boldsymbol{W}_{t-1}$
  while not converged for e_t **do**
    $\Delta \boldsymbol{W} \leftarrow \boldsymbol{W}_t - \boldsymbol{W}_{\text{down}}$
    Determine the $(1-k)$-th quantile threshold ψ_k of elements in $\{|\Delta \boldsymbol{W}_{i,j}|\}$
    Compute risk mask $\boldsymbol{\mathcal{M}}_{\text{risk}} \leftarrow \left[\mathbb{I}\left(|\Delta \boldsymbol{W}_{i,j}| > \psi_k\right)\right]_{i,j}$ ▷ Eq. 5
    Compute loss $\mathcal{L}$ for e_t
    Calculate raw gradient $\nabla_{\boldsymbol{W}_t} \mathcal{L}$
    if $t = 1$ **then**
      Use the raw gradient for the update
    else if $\boldsymbol{Q}^{(t-1)}$ is empty **then**
      Project gradient: $\nabla_{\boldsymbol{W}_t} \mathcal{L} \leftarrow \nabla_{\boldsymbol{W}_t} \mathcal{L} \odot \boldsymbol{\mathcal{M}}_{\text{risk}}$
    else
      Project gradient: $\nabla_{\boldsymbol{W}_t} \mathcal{L} \leftarrow \left(\nabla_{\boldsymbol{W}_t} \mathcal{L} - \nabla_{\boldsymbol{W}_t} \mathcal{L} \boldsymbol{Q}^{(t-1)} (\boldsymbol{Q}^{(t-1)})^\top\right) \odot \boldsymbol{\mathcal{M}}_{\text{risk}}$ ▷ Eq. 6
    end if
    Update weights: $\boldsymbol{W}_t \leftarrow \boldsymbol{W}_t - \eta \nabla_{\boldsymbol{W}_t} \mathcal{L}$
  end while
  Extract decoupling features $\boldsymbol{x}'_v, \boldsymbol{x}'_p$ from v_t, p_t
  for modality $* \in \{v, p\}$ **do**
    Normalize $\hat{\boldsymbol{x}}_* \leftarrow \boldsymbol{x}'_* / \|\boldsymbol{x}'_*\|_2$
    if $\boldsymbol{U}_*^{(t-1)}$ is empty **then**
      $\tilde{\boldsymbol{x}}_* \leftarrow \hat{\boldsymbol{x}}_*$
    else
      $\tilde{\boldsymbol{x}}_* \leftarrow \hat{\boldsymbol{x}}_* - \boldsymbol{U}_*^{(t-1)} (\boldsymbol{U}_*^{(t-1)})^\top \hat{\boldsymbol{x}}_*$ ▷ Eq. 2
    end if
    if $\|\tilde{\boldsymbol{x}}_*\|_2 > \epsilon$ **then**
      $\boldsymbol{U}_*^{(t)} \leftarrow [\boldsymbol{U}_*^{(t-1)}, \tilde{\boldsymbol{x}}_* / \|\tilde{\boldsymbol{x}}_*\|_2]$ ▷ Eq. 3
    else
      $\boldsymbol{U}_*^{(t)} \leftarrow \boldsymbol{U}_*^{(t-1)}$
    end if
    Compute semantic prototype $\boldsymbol{k}_*$ and update Bank $\boldsymbol{\mathcal{K}}_* \leftarrow [\boldsymbol{\mathcal{K}}_*, \boldsymbol{k}_*]$ ▷ Eq. 7
  end for
  Merge basis $\boldsymbol{U}_{\text{merged}}^{(t)} \leftarrow [\boldsymbol{U}_v^{(t)}, \boldsymbol{U}_p^{(t)}]$
  $\boldsymbol{Q}^{(t)}, \boldsymbol{R}^{(t)} \leftarrow \text{QR}(\boldsymbol{U}_{\text{merged}}^{(t)})$ ▷ Eq. 4
end for

#=== Phase 2: Inference Phase (Multi-Granularity Knowledge Integration) ===#

procedure INFERENCE($\boldsymbol{x}$)
  Extract global query vectors $\boldsymbol{q}_v, \boldsymbol{q}_p$ from input $\boldsymbol{x}$
  Compute alignment scores $s_v \leftarrow \max_j \langle \boldsymbol{q}_v, \boldsymbol{k}_{v,j} \rangle$, $s_p \leftarrow \max_j \langle \boldsymbol{q}_p, \boldsymbol{k}_{p,j} \rangle$ ▷ Eq. 8
  Determine gate $\mathcal{G} \leftarrow \mathbb{I}\left(\bigwedge_{* \in \{v,p\}} (s_* > \tau_{\min}) \wedge \left(\bigvee_{* \in \{v,p\}} (s_* > \tau) \vee |s_v - s_p| < \tau_{\text{gap}}\right)\right)$ ▷ Eq. 9
  Compute backbone output $\boldsymbol{h}_{\text{down}} \leftarrow \boldsymbol{W}_{\text{down}} \boldsymbol{x}$
  if $\mathcal{G} == 1$ **then** ▷ Route to Edited Knowledge Branch
    Compute auxiliary output $\boldsymbol{h}_t \leftarrow \boldsymbol{W}_t \boldsymbol{x}$
    Calculate perturbation $\Delta \boldsymbol{h} \leftarrow \boldsymbol{h}_t - \boldsymbol{h}_{\text{down}}$
    Construct fusion mask $\boldsymbol{\mathcal{M}}_{\text{fusion}} \leftarrow [\mathbb{I}(|\Delta \boldsymbol{h}_{i,j}| > \mathbb{E}[|\Delta \boldsymbol{h}|] + \sigma \cdot \text{Std}[|\Delta \boldsymbol{h}|])]_{i,j}$ ▷ Eq. 10
    $\boldsymbol{\mathcal{M}}_{\text{gate}} \leftarrow \mathcal{G} \cdot \boldsymbol{\mathcal{M}}_{\text{fusion}}$
    $\boldsymbol{h}_{\text{final}} \leftarrow (1 - \boldsymbol{\mathcal{M}}_{\text{gate}}) \odot \boldsymbol{h}_{\text{down}} + \boldsymbol{\mathcal{M}}_{\text{gate}} \odot \boldsymbol{h}_t$ ▷ Eq. 11
  else ▷ Route to Pre-trained Backbone
    $\boldsymbol{h}_{\text{final}} \leftarrow \boldsymbol{h}_{\text{down}}$
  end if
  return $\boldsymbol{h}_{\text{final}}$
end procedure

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately state the claims regarding mitigating catastrophic forgetting and preserving general capabilities, which are empirically validated in Section 5 and theoretically supported in Appendix B.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper explicitly discusses the theoretical capacity limits of the proposed projection-based lifelong learning method and proposes an SVD-based subspace compression solution for extreme scaling scenarios in Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Appendix B provides formal definitions, lemmas, and a complete proof for the Theorem on the Sensitivity of Knowledge Perturbation and its corollary, explicitly stating all necessary assumptions such as the small perturbation assumption.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides code in the public GitHub repository at `https://github.com/sunlitsong/ROSE` and includes the complete algorithmic implementation details along with the pseudocode of the framework in Appendix H.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The source code is publicly available at `https://github.com/sunlitsong/ROSE`, as stated in the abstract.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 and the appendix specify the benchmark datasets, base model architectures, hyperparameter settings and sensitivities, and the evaluation metrics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our evaluation strictly follows the standardized deterministic protocols of established benchmarks (MMEdit, VLKEB), computing exact aggregate metrics over the fixed test sets.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 5.5 provides a detailed analysis of the computational cost, explicitly listing the use of a single NVIDIA RTX 4090 GPU, peak VRAM consumption, and execution time per query/edit.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research aims to improve the factuality of MLLMs and safely update their knowledge, aligning strictly with the NeurIPS Code of Ethics by promoting reliable and hallucination-free AI.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We have included a dedicated Broader Impacts in Appendix G, which explicitly discusses both the positive societal impacts and the potential negative risks along with corresponding mitigation strategies.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper proposes a methodological framework and does not release new high-risk pretrained models, large generators, or scraped datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The original creators of the datasets (MMEdit, VLKEB) and base models (LLaVA, MiniGPT-4, Qwen) are appropriately cited in Section 2 and Section C.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper focuses on a new algorithmic framework and evaluates it on existing open-source benchmark datasets, rather than introducing new datasets or foundational models.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The research methodology and evaluations do not involve crowdsourcing or human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The research does not involve human subjects, making IRB approval non-applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.