KORE: Enhancing Knowledge Injection for Large Multimodal Models via Knowledge-Oriented Controls

Kailin Jiang [1 2] **Hongbo Jiang** [3] **Ning Jiang** [4] **Zhi Gao** [5 2] **Jinhe Bi** [6] **Yuchen Ren** [7] **Bin Li** [1] **Yuntao Du** [8 9]
Lei Liu✉ [1] **Qing Li**✉ [2]

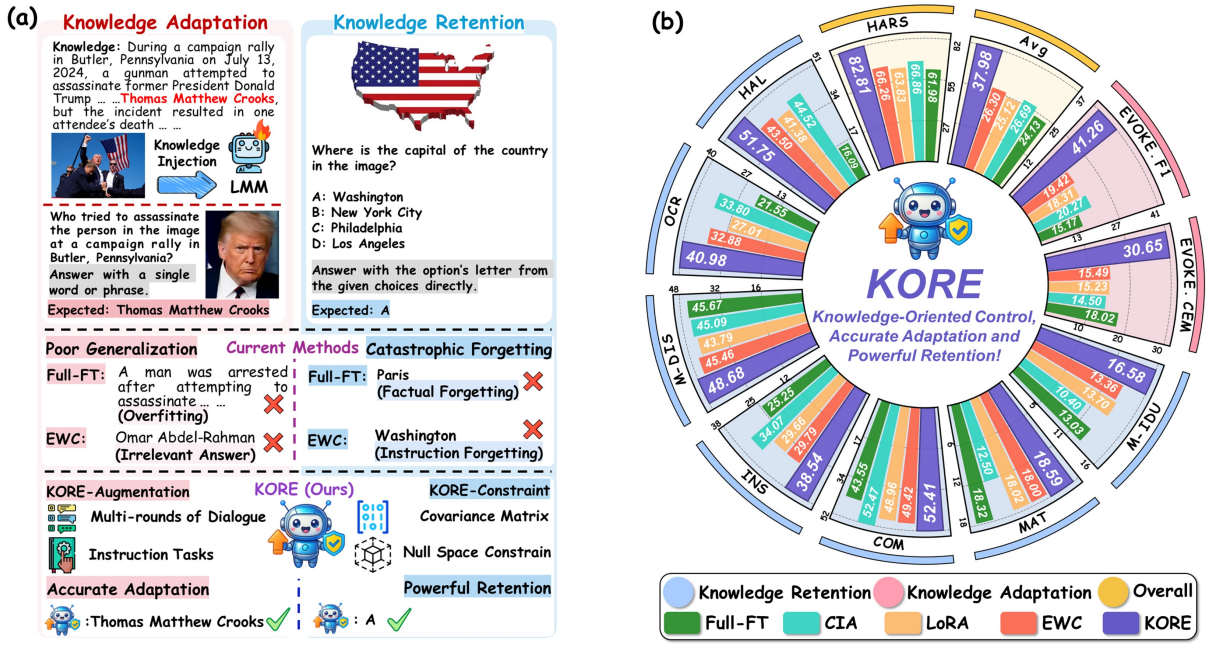

Figure 1. **(a)** Challenges of knowledge adaptation and retention in knowledge injection. While existing methods suffer from poor generalization and catastrophic forgetting, our proposed KORE leverages knowledge-oriented controls to achieve both accurate adaptation and powerful retention. **(b)** Performance of various methods on LLaVA-v1.5 (7B). Red, blue and yellow shading correspond to knowledge adaptation, knowledge retention and overall evaluations, respectively. The definition of overall evaluations' metrics are in § 4.

Abstract

Large Multimodal Models encode extensive factual knowledge in their pre-trained weights. However, its knowledge remains static and limited, unable to keep pace with real-world developments, which hinders continuous knowledge acquisition. Effective knowledge injection thus becomes critical, involving two goals: knowledge adaptation (injecting new knowledge) and knowledge retention (preserving old knowledge). Existing methods often struggle to learn new knowledge and suffer from catastrophic forgetting. To address these challenges, we propose **KORE**, a synergistic method centered around **K**n**O**wledge-o**R**ient**E**d controls. These controls are implemented through a two-stage optimization process: (1) KORE automatically converts individual knowledge items into structured and comprehensive knowledge to ensure that the model accurately learns new knowledge, enabling accurate adaptation. (2) KORE stores previous knowledge in the covari-

[1]University of Science and Technology of China [2]State Key Laboratory of General Artificial Intelligence, BIGAI [3]Xiamen University [4]Northeast Forestry University [5]Beijing Institute of Technology [6]Ludwig Maximilian University of Munich [7]The University of Sydney [8]C-FAIR&school of software, Shandong University [9]State Key Lab. for Novel Software Technology, Nanjing University, P.R. China. Correspondence to: Lei Liu <liulei13@ustc.edu.cn>, Qing Li <dylan.liqing@gmail.com>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

ance matrix of LMM's linear layer activations and initializes the adapter by projecting the original weights into the matrix's null space, defining a fine-tuning direction that minimizes interference with previous knowledge, enabling powerful retention. Extensive experiments on various LMMs, including LLaVA-v1.5 (7B), LLaVA-v1.5 (13B), and Qwen2.5-VL (7B), show that KORE achieves superior new knowledge injection performance and effectively mitigates catastrophic forgetting.

1. Introduction

Large Language Models and Large Multimodal Models demonstrate a remarkable ability to store vast world knowledge within their pre-trained weights and recall it during inference (Petroni et al., 2019; Brown et al., 2020; Roberts et al., 2020; DeepSeek-AI et al., 2024). However, their knowledge remains static and fails to keep pace with the evolving real world, leading to outdated responses and an inability to acquire new information continuously. Therefore, effective knowledge injection methods are crucial, enabling models to inject new knowledge while preserving previous knowledge (*e.g.,* knowledge adaptation and retention in Figure 1), thus supporting continuous model evolution (Ovadia et al., 2024; Mecklenburg et al., 2024).

The most direct method for injecting new knowledge is full fine-tuning, which updates all model weights. However, this strategy incurs prohibitive computational and storage costs. To address this, Parameter-Efficient Fine-Tuning (PEFT) methods have been introduced for resource-friendly adaptation. PEFT techniques, such as adding adapters (Houlsby et al., 2019; Hu et al., 2022) or new tokens (Lester et al., 2021; Sabbatella et al., 2024), drastically reduce the number of trainable parameters by freezing the original pre-trained weights. Despite their success, both full fine-tuning and PEFT methods face significant limitations. They often lead to catastrophic forgetting of pre-existing knowledge and struggle to achieve robust generalization. While full fine-tuning can minimize loss on the training data in § F, it frequently overfits, failing to effectively extract and manipulate the newly acquired knowledge (*e.g.,* Full-FT repeats training data in Figure 1).

Numerous continual learning techniques, such as rehearsal (Li & Hoiem, 2017a; Hou et al., 2019) and parameter regularization (Kirkpatrick et al., 2017; Li & Hoiem, 2017b), have been proposed to mitigate catastrophic forgetting. However, these methods often fail to balance new knowledge acquisition with prior knowledge retention. For example, regularization approaches like EWC (Kirkpatrick et al., 2017) may impair adaptation to new data, resulting in irrelevant responses and instruction forgetting (*e.g.,* EWC leads to irrelevant answer and instruction forgetting in Figure 1). Drawing inspiration from data augmentation's ability to enhance new knowledge learning (Singhal et al., 2023; Allen-Zhu & Li, 2024) and continual learning's capacity to preserve old knowledge (McCloskey & Cohen, 1989; Ratcliff, 1990), our proposed KORE optimizes the balance between injecting new knowledge and preserving old knowledge, enabling accurate adaptation and powerful retention.

Overall, KORE is a synergistic method centered around knowledge-oriented controls, implemented through a two-stage optimization process that integrates KORE-AUGMENTATION and KORE-CONSTRAINT. Unlike general augmentation techniques that produce superficial and discrete data variations, KORE automatically augments each piece of knowledge into multi-rounds of dialogue and instruction tasks data. This process constructs profound and structured knowledge, which ensures the generalization and internalization of new knowledge and enables the model to flexibly extract and manipulate learned knowledge during inference. Simultaneously, KORE stores multimodal knowledge in covariance matrix C of linear layer activations, assuming C effectively captures previous knowledge (Verification in § 3.3). We then decompose C and extract its null space. Original weights are projected into this null space to initialize a adapter for fine-tuning, which ensures a tuning direction that minimally interferes with the previous knowledge.

To validate the effectiveness of our method, we conducted extensive experiments on multiple representative LMMs. The results in Figure 1 demonstrate that KORE exhibits superior performance in both knowledge adaptation and retention compared to standard fine-tuning (*e.g.,* Full-FT, LoRA) and continual learning methods (*e.g.,* EWC, CIA). Moreover, KORE can augment arbitrary knowledge into a structured format and enables customizable knowledge constraints that can be applied based on specific retention needs (§ 4.2). By balancing adaptation and retention through knowledge-oriented controls, KORE achieves superior performance without sacrificing flexibility, highlighting its key role in efficient knowledge injection for broader application.

In summary, our main contributions include:

- We propose KORE, a synergistic method centered on knowledge-oriented controls, addressing the critical trade-off between knowledge internalization and catastrophic forgetting in LMMs during knowledge injection and achieving superior overall performance.
- To achieve *accurate adaptation*, we propose KORE-AUGMENTATION, which automatically expands individual knowledge items into multi-round dialogues and instruction-following data, and we further construct the KORE-74K dataset to serve as a new resource for future knowledge injection research.

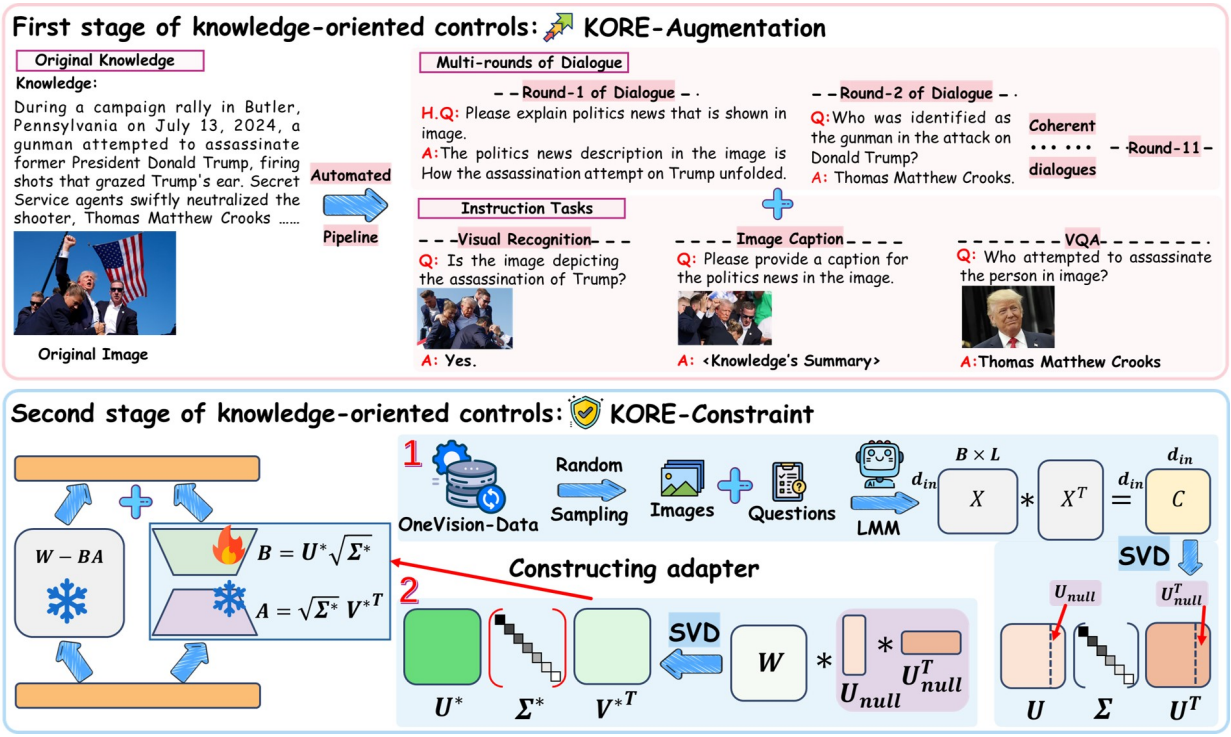

Figure 2. **Overview.** KORE is a synergistic method centered around knowledge-oriented controls, implemented through a two-stage optimization process that integrates KORE-AUGMENTATION and KORE-CONSTRAINT. **(1)** KORE-AUGMENTATION automatically converts each piece of knowledge into profound and structured knowledge. **(2)** KORE-CONSTRAINT minimizes interference with previous knowledge by initializing an adapter with null space that stores covariance matrix of previous knowledge.

- To ensure *powerful retention*, we introduce KORE-CONSTRAINT, which stores previous knowledge in covariance matrix of the model's linear layer activations and initializes the adapter by extracting covariance matrix's null space, thereby defining a fine-tuned direction that minimizes interference with previous knowledge.

2. Related Works

2.1. Knowledge Injection

Injecting new knowledge into models is a critical challenge with two main paradigms. Retrieval-Augmented Generation (Song et al., 2016; Fan et al., 2020; Lewis et al., 2020), preserves pre-trained knowledge by leveraging an external knowledge base at inference time, but its efficacy depends on the retrieval system's quality and speed. In contrast, the alternative paradigm directly modifies model parameters, often through efficient methods like full fine-tuning, parameter-efficient fine-tuning (Hu et al., 2022; Lauscher et al., 2020). However, these techniques face a dual challenge, as they often struggle to effectively inject knowledge while still causing catastrophic forgetting (Ovadia et al., 2024; Mecklenburg et al., 2024). This highlights a fundamental trade-off between knowledge adaptation and retention, which remains a core problem in knowledge injection.

2.2. Knowledge Forgetting

Knowledge injection is fundamentally a continual learning problem that focuses on acquiring new knowledge while retaining prior abilities, ensuring knowledge adaptation without catastrophic forgetting (Liu et al., 2025a; Huo & Tang, 2025; Song et al., 2025; Zheng et al., 2025). To tackle these challenges, existing continual learning methods are broadly categorized as follows. Techniques relying on parameter regularization aim to preserve the stability of the model's most critical parameters (Kirkpatrick et al., 2017; Li & Hoiem, 2017b; Feng et al., 2022; Wang et al., 2023; Liu et al., 2024a; Qiao et al., 2025; Qi et al., 2024; Qiao et al., 2025; Chen et al., 2025; Liang et al., 2025; Fang et al., 2025). Approaches focused on architecture achieve knowledge retention by introducing either parameter isolation (Mallya & Lazebnik, 2018; Serra et al., 2018; Cao et al., 2024; Zhang et al., 2025), adaptive structural elements (Yoon et al., 2018; Hung et al., 2019), or fully modular designs (Shen et al., 2019). Rehearsal-based strategies maintain previous capabilities through experience replay utilizing memory buffers (Bonicelli et al., 2022; Chen & Chang, 2023). Finally, prompt-based methods boost efficiency by employing specific learnable prompts, circumventing the need for explicit data storage (Wang et al., 2022b; Smith et al., 2023; Wang et al., 2022a).

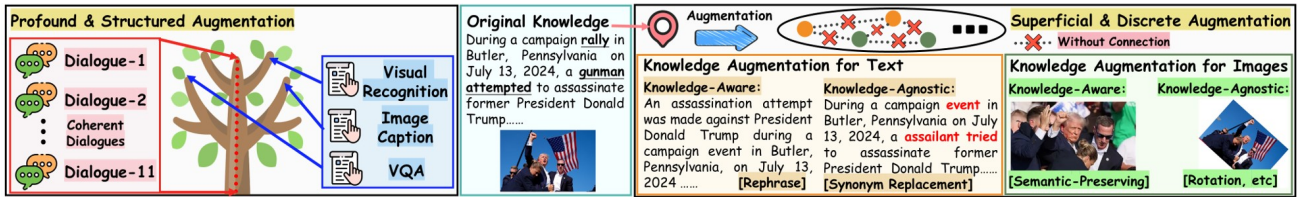

Figure 3. Comparison of KORE-AUGMENTATION **(left)** and general augmentation methods **(right)**.

3. Knowledge-Oriented Controls

3.1. Knowledge-Oriented Augmentation

Existing knowledge injection methods suffer from poor generalization and struggle to master new knowledge (Ovadia et al., 2024; Jiang et al.; Tang et al., 2025b). Inspired by recent work demonstrating that data augmentation effectively enhances generalization (Singhal et al., 2023; Allen-Zhu & Li, 2024; Wang et al., 2025; Park et al., 2025), we seek to enhance the model's ability to learn new knowledge through data augmentation. However, existing methods are limited to superficial and discrete augmentation, which is insufficient for helping models internalize new knowledge systematically. To address these limitations, we propose KORE-AUGMENTATION, a profound and structured augmentation method via automated pipeline, to build structured and comprehensive knowledge for accurate adaptation.

We observe that KORE-AUGMENTATION augments the original knowledge into multi-rounds dialogues data (forming the trunk) and instruction tasks data (forming the branches), thereby constructing a comprehensive and higher-level knowledge tree (Left part of Figure 3) that supports superior generalization and internalization of new knowledge. KORE-AUGMENTATION moves beyond enabling models to accurately fit training data for "data memorization". Instead, it focuses on helping the model comprehend and reason about the inherent logic and associations within the knowledge itself. This enables the model to think, internalize new knowledge, and effectively extract and manipulate the learned knowledge, thereby achieving real "**knowledge internalization**". In contrast, general augmentation methods are superficial and discrete. As shown in right part of Figure 3, for text augmentation, techniques such as knowledge-aware (*e.g.,* rephrasing) or knowledge-agnostic (*e.g.,* synonym replacement) only create isolated variations. Likewise, image augmentation, whether knowledge-aware (*e.g.,* semantic-preserving) or knowledge-agnostic (*e.g.,* rotation), operate on a surface level. These methods merely generate isolated data points without connection, superficially modifying existing knowledge to broaden exposure. They fail to construct a coherent knowledge structure. Consequently, general augmentation methods offer limited support for the generalization and internalization of new knowledge. We experimentally validate this statement in § 4.5 and KORE-AUGMENTATION's implementation details are as follows:

Part 1: Constructing Multi-rounds of Dialogue Data. The multi-rounds of dialogue data for each knowledge sample consists of two components: heuristic Q&A (H.Q in Figure 2) and dialogue Q&A. ❶ The heuristic Q&A is constructed randomly using manually written templates. ❷ For dialogue Q&A, we design rigorous rules and diverse task examples, using GPT-4o to generate up to 10 dialogues from original textual knowledge. Ultimately, this process yields 75,710 dialogue data. **Part 2: Constructing Instruction Tasks Data.** We use News's titles or Entity's names as search key words to retrieve the top five images via Google Search. Visual features of both original and collected images are extracted using CLIP (Radford et al., 2021). The two images with the highest cosine similarity are retained. ❶ **Visual Recognition:** For this task, questions are randomly selected from a manually written template, and the answer is defined as "Yes". One of the previously retained images serves as the query image, accompanied by the instruction, *"Answer this question with Yes or No"*. ❷ **Image Caption:** For this task, answer is a summary generated by GPT-4o based on original textual knowledge. Question is randomly selected from templates, and query image is those remaining from previous steps. And instruction is *"Answer this question in one paragraph"*. ❸ **VQA:** First, we utilize GPT-4o to generate quadruplets (Q, A, S, H) from original textual knowledge, where Q and A form a question-answer pair, S is the subject in question and H is hypernym corresponding to the subject. Subsequently, the subject and hypernym are combined to form a search key words for retrieving and downloading images from Google. The instruction is: *"Answer the question using a single word or phrase"*. This process yields 46,468 VQA samples.

Ultimately, We construct KORE-74K using original knowledge of EVOKE, and KORE is training on KORE-74K. See more details about KORE-AUGMENTATION in § G.

3.2. Knowledge-Oriented Constraint

Large Multimodal Models effectively leverage their pre-trained knowledge to perform a wide range of tasks, and these capabilities are reflected as distinct patterns within their internal activation covariance matrices (Meng et al., 2023; Yang et al., 2024). However, integrating new knowl-

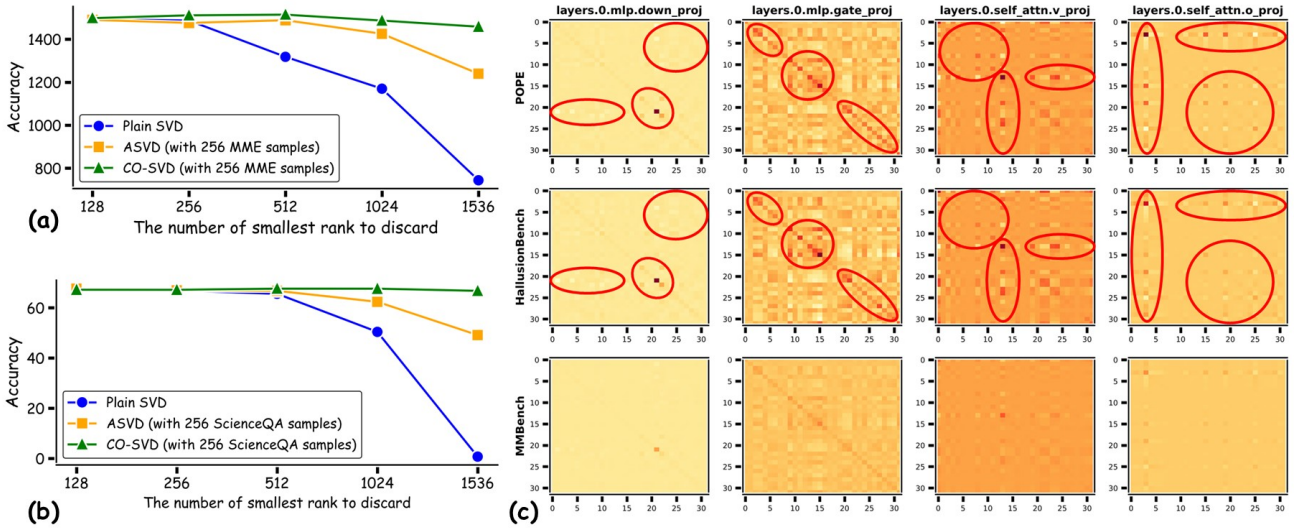

Figure 4. Performance (higher is better) on **(a)** MME and **(b)** ScienceQA after reconstruction. **(c)** Covariance matrix visualization for 4 different input activations in the 0-th block. We down-sample the heatmaps into 32×32. Similar patterns are marked in red circles.

edge or skills into these models presents a fundamental challenge. Direct fine-tuning, the conventional approach, often disrupts these carefully established internal structures, leading to the catastrophic forgetting of prior abilities (Rebuffi et al., 2016; Shi et al., 2024). Consequently, the field of continual learning has focused on developing various coSnstraint-based methods to mitigate this performance degradation and preserve foundational knowledge during adaptation.

Inspired by this, we propose KORE-CONSTRAINT, a knowledge-oriented constraint method. It stores previous knowledge in covariance matrix of activations from LMM's linear layers, decomposes this matrix to obtain its null space, and projects the original weights onto this subspace to initialize adapter. This process ensures that the fine-tuning direction minimally interferes with previous knowledge.

Following prior work (Meng et al., 2023; Fang et al., 2025; Tang et al., 2025a), we collect activations from LMMs on a set of random samples representing pre-trained knowledge. Let the input activations to a linear layer be $X \in \mathbb{R}^{d_{in} \times BL}$, where B is the number of samples, L is the sequence length, and d_{in} is the input dimension. And its covariance be $C = XX^T \in \mathbb{R}^{d_{in} \times d_{in}}$. Given pre-trained weights W_0, the fine-tuned weights through LoRA are given by: $W^* = W_0 + BA$. To achieve knowledge retention, we want to ensure the output activations derived from pretrained knowledge remain consistent after fine-tuning, formalized by the following condition: $W^*C = (W_0 + BA)C \approx W_0C$. Simplifying this equation further, we obtain: $BAC \approx \mathbf{0}$, and to solve this problem, our goal is to have A located in the null space matrix (Wang et al., 2021) of C, which is formulated as $AC = \mathbf{0}$. Following the existing methods for conducting null space projection, we first apply a Singular Value Decomposition (SVD) to $C = XX^T$:

$$\text{SVD}\left(XX^T\right) = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{u}_i^T, \tag{1}$$

where U is the orthogonal matrix of left singular vectors, respectively, and Λ is a diagonal matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_R > 0$ (with $R = \text{rank}(C)$). The null space of C is spanned by $U_{\text{null}} \in \mathbb{R}^{d_{\text{in}} \times (d_{\text{in}} - R)}$, a submatrix containing the last $(d_{\text{in}} - R)$ columns of U that correspond to zero singular values. As shown in the first step on the right side of Figure 2, U_{null} satisfies $U_{\text{null}}^T C = \mathbf{0}$.

We approximate the null space with $\hat{U} \in \mathbb{R}^{d_{\text{in}} \times r}$, a submatrix containing the r left singular vectors from U associated with the smallest singular values, where r is the predefined LoRA's rank. From this, we define a knowledge-oriented constraint projector $P = \hat{U}\hat{U}^T$. As shown in Figure 2, we then initialize the LoRA adapters by factorizing the pre-trained weights projected into this null space. We compute the SVD of the projected weights: $\text{SVD}\left(W_0 P\right) = \left\{U^*, \Sigma^*, (V^*)^T\right\}$ and initialize the adapter matrices B and A as:

$$B = U^* \sqrt{\Sigma^*}, \qquad A = \sqrt{\Sigma^*}(V^*)^T, \tag{2}$$

where $\sqrt{\Sigma^*}$ denotes the diagonal matrix with entries for singular values. Finally, to ensure the model is unchanged at the start of fine-tuning, the original weight matrix is adjusted with a residual term:

$$W_0' = W_0 - BA. \tag{3}$$

Given the asymmetry between A and B, fine-tuning only B suffices for strong performance (Zhang et al., 2023; Zhu et al., 2024). By freezing A within the null space of C,

Table 1. **Performance of KORE compared with nine baseline methods.** Row of "LLaVA-v1.5 (7B)" shows performance of pre-trained model. **Bold** and underline denote the top and runner-up scores, respectively. Results with gray texture are excluded from sorting.

Method	Params	EVOKE		COM ↑	OCR ↑	M-DIS ↑	INS ↑	M-IDU ↑	MAT ↑	HAL ↑	K.A ↑	K.R ↑	Avg ↑	HARS ↑
		CEM ↑	F1↑											
LLaVA-v1.5 (7B)	—	4.89	9.34	65.61	45.59	49.22	66.33	26.37	19.33	54.32	7.12	46.74	26.93	—
Full-FT	6,759M	18.02	15.17	43.55	21.55	45.67	25.25	13.03	18.32	16.09	16.60	31.66	24.13	61.98
LoRA	340M	15.23	18.31	48.96	27.01	43.79	29.66	13.70	18.02	41.38	16.77	33.47	25.12	63.83
Replay	340M	11.36	17.98	59.72	37.98	48.64	**62.33**	**19.31**	19.17	51.67	14.67	**43.00**	28.83	66.04
EWC	340M	15.49	19.42	49.42	32.88	45.46	29.79	13.36	18.00	43.50	17.46	35.14	26.30	66.26
LwF	340M	14.58	19.99	53.14	28.77	43.41	36.19	13.68	18.22	44.18	17.29	35.44	26.36	66.26
MoELoRA	340M	6.45	12.20	60.79	38.79	48.27	35.03	17.85	19.79	49.99	9.33	40.51	24.92	37.22
O-LoRA	340M	6.44	12.08	**61.47**	40.91	48.07	34.85	17.28	**19.87**	51.12	9.26	41.25	25.26	36.70
SEFE	340M	16.22	24.12	36.77	35.94	48.55	42.90	10.55	18.59	31.55	20.17	33.02	26.60	67.56
CIA	340M	14.50	20.27	52.47	33.80	45.09	34.07	10.40	12.50	44.52	17.39	35.99	26.69	66.86
KORE (rank=235)	340M	**30.65**	**41.26**	52.41	**40.98**	**48.68**	38.54	16.58	18.59	**51.75**	**35.96**	40.00	**37.98**	**82.81**
KORE (rank=256)	369M	31.05	41.32	52.48	39.96	48.96	60.02	23.18	18.09	51.50	36.19	42.10	39.14	84.93

KORE ensures $AC \approx \mathbf{0}$, making the update term BAC negligible regardless of any changes to B. Proof is in § C.

3.3. Analysis of Knowledge-Oriented Constraint

KORE-CONSTRAINT relies on the premise that the extracted covariance matrix effectively captures knowledge from previous data. Therefore, we expand CO-SVD (Yang et al., 2024) from pure text scenarios to multimodal scenarios to verify *"whether covariance matrices can capture multimodal knowledge and activate distinct modes?"* We apply Plain SVD, ASVD (Yuan et al., 2023) and CO-SVD to fully decompose all layers of LLaVA-v1.5 (7B) pre-trained weights. The weights are reconstructed by removing the components corresponding to the r smallest singular values.

Our analysis reveals two key findings: ❶ Figure 4 (a) and (b) demonstrate that CO-SVD exhibits superior performance retention compared to Plain SVD, ASVD and suggest that multimodal knowledge can be effectively captured and stored in covariance matrix. ❷ Figure 4 (c) shows that covariance matrices of linear layer inputs share similar outlier patterns for related tasks (*e.g.,* POPE and HallusionBench), but differ from unrelated ones (*e.g.,* MMBench), indicating that distinct tasks exhibit different outlier distributions in the covariance matrix. To build a multi-dimensional covariance matrix for KORE, we finally sample 256 examples from OneVision (Li et al., 2025) (General, Doc/Chart/Screen, Math/Reasoning, General OCR). See more results in § D.

4. Experiments

In this section, we introduce experimental content. See more details and evaluation protocol in § B.

Setup 1: Knowledge Adaptation Evaluation. we evaluate LMMs by fine-tuning them on EVOKE (Jiang et al.), where knowledge is injected as an image-text pair, with evaluation questions derived from the text.

Setup 2: Knowledge Retention Evaluation. We evaluate fine-tuned LMMs on 12 benchmarks across 7 capability dimensions, covering the following tasks: ❶ Comprehensive Evaluation (COM): MME (Fu et al., 2025) and MMBench (Liu et al., 2024b); ❷ Optical Character Recognition (OCR): SEEDBench2_Plus (Li et al., 2024) and OCRVQA (Mishra et al., 2019); ❸ Multidisciplinary Reasoning (M-DIS): ScienceQA (Lu et al., 2022) and MMMU (Yue et al., 2023); ❹ Instruction Following (INS): MIA-Bench (Qian et al., 2025); ❺ Multi-Turn Multi-Image Dialog Understanding (M-IDU): MMDU (Liu et al., 2025b); ❻ Mathematical Reasoning (MAT): MathVista (Lu et al., 2024) and MathVision (Wang et al., 2024); ❼ Hallucination (HAL): POPE (Li et al., 2023) and HallusionBench (Guan et al., 2024).

Setup 3: Metrics for Overall Evaluations. ❶ ***Avg*** score is the average performance of knowledge adaptation and retention (*i.e.,* K.A and K.R). ❷ ***HARS*** denotes the Harmonized Adaptation-Retention Score, which draws an analogy to the balance between Precision and Recall by integrating knowledge adaptation and retention into a unified metric to evaluate their equilibrium through the following formula:

$$HARS = 2 \cdot \frac{f_A \cdot f_R}{f_A + f_R} \tag{4}$$

where $f_A = \frac{G_A}{G_A+100} \times 100$ and $f_R = G_R + 100$ denote the normalized scores for adaptation and retention, respectively. These scores are derived from the relative gains $G_A = (K.A - K.A_0)/K.A_0$ and $G_R = (K.R - K.R_0)/K.R_0$, where $K.A_0$ and $K.R_0$ denote knowledge adaptation and retention performance of the pre-trained model.

Setup 4: Baseline Methods. We compare against several baselines: Full-FT, LoRA, Replay, EWC, LwF (Li & Hoiem, 2017b), MoELoRA (Luo et al., 2024), O-LoRA (Wang et al., 2023), SEFE (Chen et al., 2025) and CIA (Qiao et al., 2025).

Table 2. Performance comparison between KORE and baseline methods on fine-grained knowledge retention evaluations. MM^B: MMBench; $SEED^{B2P}$: SEEDBench2_Plus; $Math^T$: MathVista ; $Math^I$: MathVision; $Hall^B$: HallusionBench. MME's score is normalized.

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^B ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	MMMU ↑	MIA^B ↑	MMDU ↑	$Math^T$ ↑	$Math^I$ ↑	POPE ↑	$Hall^B$ ↑	
LLaVA-v1.5 (7B)	66.63	64.60	38.78	52.41	69.83	28.60	66.33	26.37	25.50	13.16	86.87	21.76	46.74
Full-FT	34.17	52.92	31.44	11.65	67.13	24.20	25.25	13.03	24.70	11.94	74.22	9.27	31.66
LoRA	44.06	53.87	30.22	23.80	66.18	21.40	29.66	13.70	23.20	12.83	73.97	8.78	33.47
Replay	58.96	60.48	38.34	37.73	68.77	28.50	**62.33**	**19.31**	25.20	13.13	**85.44**	17.90	**43.00**
EWC	48.57	50.26	33.60	32.16	65.71	25.20	29.79	13.36	23.30	12.76	76.22	10.77	35.14
LwF	50.87	55.41	32.02	25.52	66.21	20.60	36.19	13.68	24.40	12.04	79.23	9.13	35.44
MoELoRA	58.26	**63.32**	37.42	40.17	**69.04**	27.50	35.03	17.85	27.80	11.78	80.70	19.29	40.51
O-LoRA	**60.30**	62.63	37.90	43.91	68.84	27.30	34.85	17.28	**28.20**	11.55	81.46	20.78	41.25
SEFE	17.27	56.27	**39.00**	32.88	66.10	**31.00**	42.90	10.55	23.90	**13.29**	50.61	12.48	33.02
CIA	50.65	54.30	33.29	34.31	67.28	22.90	34.07	10.40	24.00	11.60	79.29	9.75	35.99
KORE (rank=235)	49.84	54.98	37.73	**44.24**	68.06	29.30	38.54	16.58	25.10	12.09	80.99	**22.51**	40.00
KORE (rank=256)	50.06	54.90	36.89	43.03	68.51	29.40	60.02	23.18	24.70	11.48	80.77	22.23	42.10

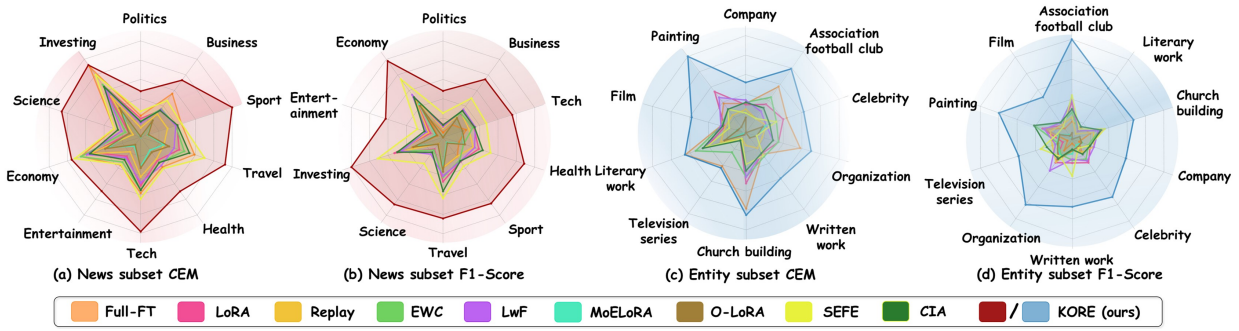

Figure 5. Comparison between KORE and baseline methods on fine-grained knowledge types.

4.1. Analysis of main results

We provide the performance of various methods in Table 1 and more results, case studies in § E and § H, from which we draw the following observations: ❶ KORE enables accurate adaptation for effectively injecting new knowledge. Specifically, KORE (rank=235) achieves improvements of 12.63 ↑ in CEM and 17.14 ↑ in F1-Score over the best baseline on EVOKE, even outperforming LoRA by more than twofold in K.A. ❷ KORE enables powerful retention for effectively preserving previous knowledge. Specifically, KORE (rank=235) outperforms Full-FT, LoRA, EWC, LwF, SEFE, CIA in K.R and achieves top scores on OCR, M-DIS, HAL. ❸ KORE achieves remarkable holistic performance by harmonizing the dual objectives of knowledge injection. Specifically, KORE (rank=235) achieves improvements of 9.15 ↑ in Avg and 15.25 ↑ in HARS over the best baseline, demonstrating its superior overall performance.

4.2. Analysis of fine-grained results

According to the fine-grained knowledge adaptation and retention performance shown in Table 2 and Figure 5, we draw the following observations: ❹ KORE achieves competitive knowledge retention. Specifically, KORE outperforms LoRA (*e.g.,* 6.53 ↑ in Avg) and continual learning methods (*e.g.,* EWC, LwF, SEFE and CIA), achieving superior scores on OCR^{VQA}, MMMU and $Hall^B$. Furthermore, by adjusting trainable parameters (rank=256) and covariance matrix source (Table 13), it closely matches or even exceeds Replay. ❺ KORE demonstrates superior performance across a wide spectrum of fine-grained knowledge. Figure 5 compares 20 fine-grained News and Entity types from EVOKE. KORE consistently outperforms all baselines, demonstrating strong and comprehensive knowledge adaptation.

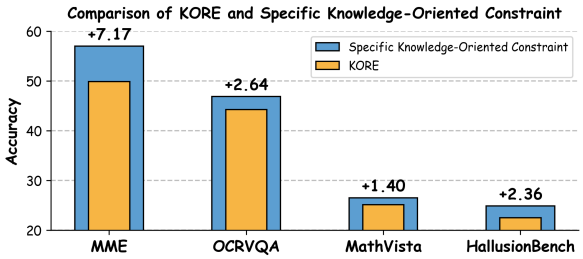

Figure 6. Performance comparison of corresponding tasks under specific knowledge-oriented constraints.

Given the diverse prior knowledge of LMMs, we investigate ***whether KORE can preserve specific knowledge without compromising new knowledge injection or other existing abilities?*** We construct specific constraints by sampling 256 data per benchmark across four dimensions. ❻ Specific constraints enhance specific knowledge retention. Figure 6 shows that specific constraints enhance targeted knowledge retention, notably with a 7.17 ↑ gain on MME, demonstrat-

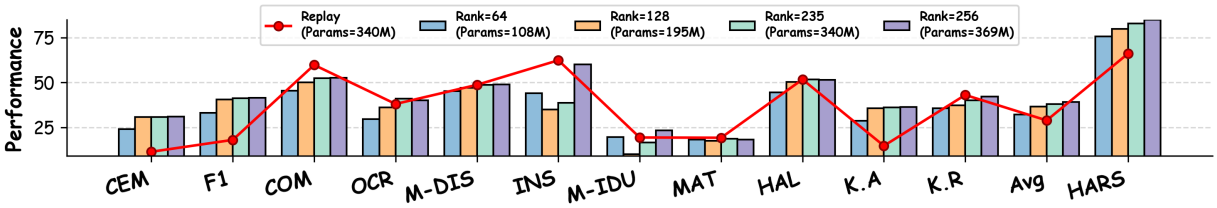

Figure 7. Comparison of different ranks for KORE with LLaVA-v1.5 (7B).

Table 3. Performance comparison between KORE and baseline methods with various LMMs scales and architectures.

Method	K.A ↑	K.R ↑	Avg ↑	HARS ↑
LLaVA-v1.5 (13B)	7.70	49.51	28.60	—
LoRA	19.55	32.64	26.09	63.15
Replay	16.13	**45.69**	30.91	66.73
KORE	**38.68**	45.35	**42.01**	**85.46**
Qwen2.5-VL (7B)	12.34	66.89	39.61	—
LoRA	14.29	38.16	26.22	22.03
Replay	15.12	**63.94**	39.53	30.89
KORE	**27.14**	58.31	**42.72**	**67.10**

Table 4. Comparison of ablation experiment results of KORE.

Method	K.A ↑	K.R ↑	Avg ↑	HARS ↑
KORE	35.96	40.00	**37.98**	**82.81**
W/o Augmentation	14.57	**40.16**	27.37	64.14
W/o Constraint	**38.82**	35.78	37.30	79.04
W/o Frozen Matrix A	36.85	38.92	37.88	81.96

ing their potential for tailored knowledge retention.

4.3. Analysis of various LMM scales and architectures

We evaluate the universality and robustness of KORE on larger and architecturally distinct models, using Replay and LoRA as baseline methods. And we draw the following observations: ❼ KORE shows enhanced superiority on a larger-scale LMM. Table 3 shows that KORE surpasses LoRA across all dimensions, achieving improvements of 11.10 ↑ in Avg and 18.73 ↑ in HARS over Replay. ❽ KORE 's effectiveness is not architecture-specific. On Qwen2.5-VL (7B), KORE achieves improvements of 3.19 ↑ in Avg and 36.21 ↑ in HARS over Replay (proving the necessity of HARS metric), demonstrating that it consistently delivers superior performance across various architectures.

4.4. Analysis of ablation experiments

In this section, we conduct extensive ablation studies (*e.g.,* Rank, W/o Augmentation, W/o Constraint and W/o Frozen Matrix A) to validate the effectiveness of KORE's design.

❾ Larger rank enhance KORE's performance. Figure 7 shows a clear trend: KORE's performance increases with higher rank and more trainable parameters on nearly all evaluations. KORE (rank=64) still surpasses Replay in K.A, Avg and HARS, only using less than half of parameters of Replay. ❿ Ablation studies reveals the effectiveness of KORE's design. Table 4 validates the design of KORE, demonstrating that the two-stage optimization design centered on knowledge-oriented controls contributes positively to overall performance. W/o Augmentation is particularly detrimental to knowledge adaptation (21.39 ↓ in K.A). Meanwhile, W/o Constraint and W/o Frozen Matrix A impairs knowledge retention.

Table 5. Performance comparison of augmentation methods.

Method	K.A ↑	K.R ↑	Avg ↑	HARS ↑
KORE-AUGMENTATION	**38.82**	**35.78**	**37.30**	**79.04**
Augmentation for Text				
Knowledge-Aware	20.29	34.86	27.58	69.43
Knowledge-Agnostic	15.60	35.71	25.65	63.54
Augmentation for Images				
Knowledge-Aware	18.33	34.02	26.17	66.48
Knowledge-Agnostic	18.33	32.09	25.21	64.71

4.5. Comparison with general augmentation methods

This section validates our claim from § 3.1 that KORE-AUGMENTATION is superior to general augmentation methods. As shown in Table 5, KORE-AUGMENTATION outperforms general augmentation methods across all metrics, notably achieving improvements of 18.53 ↑ in K.A, 9.72 ↑ in Avg and 9.61 ↑ in HARS over the best baseline. Since both KORE-AUGMENTATION and general augmentation for text utilize GPT-4o, the substantial performance gap proves that KORE-AUGMENTATION 's effectiveness stems from its superior design rather than mere distillation from GPT-4o.

5. Conclusion

In this work, we propose KORE, a synergistic method centered on knowledge-oriented controls that addresses the critical trade-off between injecting new knowledge and preserving existing knowledge. Specifically, KORE automatically converts each piece of knowledge into a more profound and structured format, ensuring the model accurately

learns and adapts to new knowledge. Simultaneously, it minimizes interference with previous knowledge by initializing an adapter with null space that stores the covariance matrix of previous knowledge, enabling powerful retention. KORE's robust performance is architecture-agnostic and exhibits enhanced superiority on larger-scale LMMs.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgement

This research is supported by Smart-Grid National Science and Technology Major Project (Grant No. 2025ZD0805500).

References

Allen-Zhu, Z. and Li, Y. Physics of language models: Part 3.1, knowledge storage and extraction. In *International Conference on Machine Learning*, 2024.

Bonicelli, L., Boschini, M., Porrello, A., Spampinato, C., and Calderara, S. On the effectiveness of lipschitz-driven rehearsal in continual learning. *Advances in Neural Information Processing Systems*, 35:31886–31901, 2022.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T. J., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., teusz Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In *NeurIPS*, 2020.

Cao, X., Lu, H., Huang, L., Liu, X., and Cheng, M.-M. Generative multi-modal models are good class incremental learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28706–28717, 2024.

Chan, C.-M., Xu, C., Yuan, R., Luo, H., Xue, W., Guo, Y.-T., and Fu, J. Rq-rag: Learning to refine queries for retrieval augmented generation. *arXiv preprint arXiv:2404.00610*, 2024.

Chen, C., Zhu, J., Luo, X., Shen, H., Gao, L., and Song, J. Coin: A benchmark of continual instruction tuning for multimodel large language models. *Advances in Neural Information Processing Systems*, 37:57817–57840, 2024.

Chen, J., Cong, R., Zhao, Y., Yang, H., Hu, G., Ip, H. H. S., and Kwong, S. SEFE: superficial and essential forgetting eliminator for multimodal continual instruction tuning. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net, 2025.

Chen, X. and Chang, X. Dynamic residual classifier for class incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18743–18752, 2023.

DeepSeek-AI, Liu, A., Feng, B., Xue, B., Wang, B.-L., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Guo, D., Yang, D., Chen, D., Ji, D.-L., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Zhang, H., Ding, H., Xin, H., Gao, H., Li, H., Qu, H., Cai, J. L., Liang, J., Guo, J., Ni, J., Li, J., Wang, J., Chen, J., Chen, J., Yuan, J., Qiu, J., Li, J., Song, J.-M., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Xu, L., Xia, L., Zhao, L., Wang, L., Zhang, L., Li, M., Wang, M., Zhang, M., Zhang, M., Tang, M., Li, M., Tian, N., Huang, P., Wang, P., Zhang, P., Wang, Q., Zhu, Q., Chen, Q., Du, Q., Chen, R. J., Jin, R., Ge, R., Zhang, R., Pan, R., Wang, R., Xu, R., Zhang, R., Chen, R., Li, S. S., Lu, S., Zhou, S., Chen, S., Wu, S.-P., Ye, S., Ma, S., Wang, S., Zhou, S., Yu, S., Zhou, S., Pan, S., Wang, T., Yun, T., Pei, T., Sun, T., Xiao, W., Zeng, W., Zhao, W., An, W., Liu, W., Liang, W., Gao, W., xuan Yu, W., Zhang, W., Li, X. Q., Jin, X., Wang, X., Bi, X., Liu, X., Wang, X., Shen, X.-C., Chen, X., Zhang, X., Chen, X., Nie, X., Sun, X., Wang, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yu, X., Song, X., Shan, X., Zhou, X., Yang, X., Li, X., Su, X., Lin, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhu, Y. X., Zhang, Y., Xu, Y., Huang, Y., Li, Y., Zhao, Y., Sun, Y., Li, Y., Wang, Y., Yu, Y., Zheng, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Tang, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y.-B., Liu, Y., Guo, Y., Wu, Y., Ou, Y., Zhu, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Zha, Y., Xiong, Y., Ma, Y., Yan, Y., Luo, Y.-W., mei You, Y., Liu, Y., Zhou, Y., Wu, Z. F., Ren, Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Huang, Z., Zhang, Z., Xie, Z., guo Zhang, Z., Hao, Z., Gou, Z., Ma, Z., Yan, Z., Shao, Z., Xu, Z., Wu, Z., Zhang, Z., Li, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z.-A., Xie, Z., Song, Z., Gao, Z., and Pan, Z. Deepseek-v3 technical report. *ArXiv*, abs/2412.19437, 2024.

Du, Y., Jiang, K., Gao, Z., Shi, C., Zheng, Z., Qi, S., and Li, Q. Mmke-bench: A multimodal editing benchmark for diverse visual knowledge. *ArXiv*, abs/2502.19870, 2025.

Duan, H., Yang, J., Qiao, Y., Fang, X., Chen, L., Liu, Y., Dong, X., Zang, Y., Zhang, P., Wang, J., Lin, D., and Chen, K. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the*

32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024, pp. 11198–11201. ACM, 2024.

Fan, Z., Gong, Y., Wei, Z., Wang, S., Huang, Y., Jiao, J., Huang, X.-J., Duan, N., and Zhang, R. An enhanced knowledge injection model for commonsense generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 2014–2025, 2020.

Fang, J., Jiang, H., Wang, K., Ma, Y., Shi, J., Wang, X., He, X., and Chua, T. Alphaedit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Feng, T., Wang, M., and Yuan, H. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9427–9436, 2022.

Fu, B., Du, Y., Chang, C., Jin, B., Deng, W., Xu, M., Yan, H., Song, W., and Wan, Y. Mmku-bench: A multimodal update benchmark for diverse visual knowledge. *arXiv preprint arXiv:2603.15117*, 2026.

Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., Ji, R., Shan, C., and He, R. Mme: A comprehensive evaluation benchmark for multimodal large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025.

Guan, T., Liu, F., Wu, X., Xian, R., Li, Z., Liu, X., Wang, X., Chen, L., Huang, F., Yacoob, Y., Manocha, D., and Zhou, T. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024.

Guo, H., Zhu, F., Zhao, H., Zeng, F., Liu, W., Ma, S., Wang, D.-H., and Zhang, X.-Y. Mcitlib: Multimodal continual instruction tuning library and benchmark. *arXiv preprint arXiv:2508.07307*, 2025.

Hou, S., Pan, X., Loy, C. C., Wang, Z., and Lin, D. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 831–839, 2019.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 2790–2799. PMLR, 2019.

Hu, E. J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Hung, S. C. Y., Tu, C.-H., Wu, C.-E., Chen, C.-H., Chan, Y.-M., and Chen, C.-S. Compacting, picking and growing for unforgetting continual learning. *ArXiv*, abs/1910.06562, 2019.

Huo, Y. and Tang, H. When continue learning meets multimodal large language model: A survey. *arXiv preprint arXiv:2503.01887*, 2025.

Jia, Y., Du, Y., Jiang, K., Liang, Y., Ren, Q., Xin, Y., Yang, R., Feng, F., Chen, M., Lu, H., et al. Benchmarking multimodal knowledge conflict for large multimodal models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 22283–22291, 2026.

Jiang, K., Du, Y., Ding, Y., Ren, Y., Jiang, N., Gao, Z., Zheng, Z., Liu, L., Li, B., and Li, Q. When large multimodal models confront evolving knowledge: Challenges and explorations. In *The Fourteenth International Conference on Learning Representations*.

Jiang, K., Jiang, N., Du, Y., Ren, Y., Li, Y., Gao, Y., Bi, J., Ma, Y., Li, B., Liu, L., and Li, Q. Mined: Probing and updating with multimodal time-sensitive knowledge for large multimodal models. *ArXiv*, abs/2510.19457, 2025.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N. C., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Lauscher, A., Majewska, O., Ribeiro, L. F. R., Gurevych, I., Rozanov, N., and Glavavs, G. Common sense or world knowledge? investigating adapter-based knowledge injection into pretrained transformers. *arXiv preprint arXiv:2005.11787*, 2020.

Lester, B., Al-Rfou, R., and Constant, N. The power of scale for parameter-efficient prompt tuning. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Kuttler, H., Lewis, M., tau Yih, W.,

Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474, 2020.

Li, B., Ge, Y., Chen, Y., Ge, Y., Zhang, R., and Shan, Y. Seed-bench-2-plus: Benchmarking multimodal large language models with text-rich visual comprehension. *arXiv preprint arXiv:2404.16790*, 2024.

Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., and Li, C. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research (TMLR)*, 2025.

Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W. X., and rong Wen, J. Evaluating object hallucination in large vision-language models. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017a.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017b.

Liang, J., Huang, W., Wan, G., Yang, Q., and Ye, M. Lorasculpt: Sculpting lora for harmonizing general and specialized knowledge in multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26170–26180, 2025.

Liu, X., Zhai, J.-T., Bagdanov, A. D., Li, K., and Cheng, M.-M. Task-adaptive saliency guidance for exemplar-free class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23954–23963, 2024a.

Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., Chen, K., and Lin, D. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pp. 216–233. Springer, 2024b.

Liu, Y., Hong, Q., Huang, L., Gomez-Villa, A., Goswami, D., Liu, X., van de Weijer, J., and Tian, Y. Continual learning for vlms: A survey and taxonomy beyond forgetting. *arXiv preprint arXiv:2508.04227*, 2025a.

Liu, Z., Chu, T., Zang, Y., Wei, X., wen Dong, X., Zhang, P., Liang, Z., Xiong, Y., Qiao, Y., Lin, D., and Wang, J. Mmdu: A multi-turn multi-image dialog understanding benchmark and instruction-tuning dataset for lvlms. *Advances in Neural Information Processing Systems*, 37: 8698–8733, 2025b.

Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

Lu, P., Bansal, H., Xia, T., Liu, J., yue Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *The Twelfth International Conference on Learning Representations*, 2024.

Luo, T., Lei, J., Lei, F., Liu, W., He, S., Zhao, J., and Liu, K. Moelora: Contrastive learning guided mixture of experts on parameter-efficient fine-tuning for large language models. *arXiv preprint arXiv:2402.12851*, 2024.

Mallya, A. and Lazebnik, S. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24: 109–165, 1989.

Mecklenburg, N., Lin, Y., Li, X., Holstein, D., Nunes, L. O., Malvar, S., Silva, B., Chandra, R., Aski, V., Yannam, P. K. R., Aktas, T., and Hendry, T. Injecting new knowledge into large language models via supervised fine-tuning. *arXiv preprint arXiv:2404.00213*, 2024.

Meng, K., Sharma, A. S., Andonian, A. J., Belinkov, Y., and Bau, D. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*. OpenReview.net, 2023.

Mishra, A., Shekhar, S., Singh, A. K., and Chakraborty, A. Ocr-vqa: Visual question answering by reading text in images. *International Conference on Document Analysis and Recognition*, pp. 947–952, 2019.

Ovadia, O., Brief, M., Mishaeli, M., and Elisha, O. Fine-tuning or retrieval? comparing knowledge injection in llms. *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP 2024)*, 2024.

Park, C. F., Zhang, Z., and Tanaka, H. New news: System-2 fine-tuning for robust integration of new knowledge. *arXiv preprint arXiv:2505.01812*, 2025.

Peng, T., Du, Y., Ji, P., Dong, S., Jiang, K., Ma, M., Tian, Y., Bi, J., Li, Q., Du, W., Xiao, F., and Cui, L. Can visual input be compressed? a visual token compression benchmark for large multimodal models. *ArXiv*, abs/2511.02650, 2025.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P. S. H., Bakhtin, A., Wu, Y., and Miller, A. H. Language models as knowledge bases? In *EMNLP/IJCNLP (1)*, pp. 2463–2473. Association for Computational Linguistics, 2019.

Qi, S., Yang, B., Jiang, K., Wang, X., Li, J., Zhong, Y., Yang, Y., and Zheng, Z. In-context editing: Learning knowledge from self-induced distributions. *ArXiv*, abs/2406.11194, 2024.

Qian, Y., Ye, H., Fauconnier, J., Grasch, P., Yang, Y., and Gan, Z. Mia-bench: Towards better instruction following evaluation of multimodal llms. In *The Thirteenth International Conference on Learning Representations*. OpenReview.net, 2025.

Qiao, J., Zhang, Z., Tan, X., Qu, Y., Ding, S., and Xie, Y. Large continual instruction assistant. In *International Conference on Machine Learning*, 2025.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, 2021.

Ratcliff, R. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285, 1990.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5533–5542, 2016.

Roberts, A., Raffel, C., and Shazeer, N. How much knowledge can you pack into the parameters of a language model? In *Conference on Empirical Methods in Natural Language Processing*, 2020.

Sabbatella, A., Ponti, A., Giordani, I., Candelieri, A., and Archetti, F. Prompt optimization in large language models. *Mathematics*, 12(6):929, 2024.

Serra, J., Suris, D., Miron, M., and Karatzoglou, A. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning*, pp. 4548–4557. PMLR, 2018.

Shen, Y., Zeng, X., and Jin, H. A progressive model to enable continual learning for semantic slot filling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 1279–1284, 2019.

Shi, H., Xu, Z., Wang, H., Qin, W., Wang, W., Wang, Y., and Wang, H. Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 58:1 – 42, 2024.

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A. K., Cole-Lewis, H., Pfohl, S., Payne, P., Seneviratne, M., Gamble, P., Kelly, C., Schärli, N., Chowdhery, A., Mansfield, P. A., y Arcas, B. A., Webster, D. R., Corrado, G. S., Matias, Y., Chou, K., Gottweis, J., Tomasev, N., Liu, Y., Rajkomar, A., Barral, J. K., Semturs, C., Karthikesalingam, A., and Natarajan, V. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Smith, J. S., Karlinsky, L., Gutta, V., Cascante-Bonilla, P., Kim, D., Arbelle, A., Panda, R., Feris, R., and Kira, Z. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11909–11919, 2023.

Song, Y., Yan, R., Li, X., Zhao, D., and Zhang, M. Two are better than one: An ensemble of retrieval-and generation-based dialog systems. *arXiv preprint arXiv:1610.07149*, 2016.

Song, Z., Yan, B., Liu, Y., Fang, M., Li, M., Yan, R., and Chen, X. Injecting domain-specific knowledge into large language models: a comprehensive survey. *arXiv preprint arXiv:2502.10708*, 2025.

Tang, P., Hu, X., Liu, Y., Ding, L., Zhang, D., Wu, X., and Zhang, D. Put the space of lora initialization to the extreme to preserve pre-trained knowledge. 2025a.

Tang, W., Cao, Y., Deng, Y., Ying, J., Wang, B., Yang, Y., Zhao, Y., Zhang, Q., Huang, X., Jiang, Y.-G., and Liao, Y. Evowiki: Evaluating llms on evolving knowledge. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025)*, 2025b.

Wang, K., Pan, J., Shi, W., Lu, Z., Ren, H., Zhou, A., Zhan, M., and Li, H. Measuring multimodal mathematical reasoning with math-vision dataset. In *Neural Information Processing Systems*, 2024.

Wang, S., Li, X., Sun, J., and Xu, Z. Training networks in null space of feature covariance for continual learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 184–193, 2021.

Wang, X., Chen, T., Ge, Q., Xia, H., Bao, R., Zheng, R., Zhang, Q., Gui, T., and Huang, X. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10658–10671, 2023.

Wang, Y., Aniri, Bi, J., Pirk, S., and Ma, Y. Ascd: Attention-steerable contrastive decoding for reducing hallucination in mllm, 2025.

Wang, Z., Zhang, Z., Ebrahimi, S., Sun, R., Zhang, H., Lee, C.-Y., Ren, X., Su, G., Perot, V., Dy, J. G., and Pfister, T. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*, pp. 631–648. Springer, 2022a.

Wang, Z., Zhang, Z., Lee, C.-Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J. G., and Pfister, T. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 139–149, 2022b.

Xu, P., Shao, W., Zhang, K., Gao, P., Liu, S., Lei, M., Meng, F., Huang, S., Qiao, Y., and Luo, P. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(3):1877–1893, 2025.

Yang, Y., Li, X., Zhou, Z., Song, S. L., Wu, J., Nie, L., and Ghanem, B. Corda: Context-oriented decomposition adaptation of large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Yoon, J., Yang, E., Lee, J., and Hwang, S. J. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations*, 2018.

Yuan, Z., Shang, Y., Song, Y., Wu, Q., Yan, Y., and Sun, G. ASVD: Activation-aware singular value decomposition for compressing large language models. In *arXiv Technical Report*, 2023.

Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., Wei, C., Yu, B., Yuan, R., Sun, R., Yin, M., Zheng, B., Yang, Z., Liu, Y., Huang, W., Sun, H., Su, Y., and Chen, W. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9556–9567, 2023.

Zhang, J., You, J., Panda, A., and Goldstein, T. Lori: Reducing cross-task interference in multi-task low-rank adaptation. *arXiv preprint arXiv:2504.07448*, 2025.

Zhang, L., Zhang, L., Shi, S., Chu, X., and Li, B. Lora-fa: Memory-efficient low-rank adaptation for large language models fine-tuning, 2023.

Zheng, J., Qiu, S., Shi, C., and Ma, Q. Towards lifelong learning of large language models: A survey. *ACM Computing Surveys*, 57(8):1–35, 2025.

Zhu, J., Greenewald, K. H., Nadjahi, K., de Ocáriz Borde, H. S., Gabrielsson, R. B., Choshen, L., Ghassemi, M., Yurochkin, M., and Solomon, J. Asymmetry in low-rank adapters of foundation models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.

A. The Use of Large Language Models in KORE

In this section, we elaborate on the precise role of large language models within KORE, as detailed below.

Usage 1: KORE-74K's construction. In § 3.1 and § G, we use GPT-4o to generate multi-rounds dialogue data, summary content of original knowledge, and quadruplets (Q, A, S, H) data, which is in line with current scientific research standards

Usage 2: Knowledge Retention Evaluation. In § 4, we employ MIA-Bench, MMDU, MathVista, and MathVision, whose evaluation requires large language models as judges, a practice consistent with current research standards.

Usage 3: Paper grammar polishing. The initial draft of the paper was written by humans and later refined for grammar using large language models, a common practice in contemporary research.

B. More details about Setup and Experimental Operation

B.1. Knowledge Adaptation Evaluation

Our knowledge adaptation evaluation completely follows the settings of EVOKE. Below we will provide a brief introduction: This paper introduces EVOKE (Jiang et al.), a new benchmark to evaluate how well Large Multimodal Models can learn evolving knowledge without forgetting their original capabilities. It reveals the limitations of current methods in knowledge adaptation and the severity of catastrophic forgetting. The study further shows that knowledge augmentation and continual learning are promising solutions, providing a framework for future research.

B.2. Knowledge Retention Evaluation

We evaluate fine-tuned LMMs' knowledge retention capabilities on 12 benchmarks across 7 capability dimensions. And we follow the settings of VLMEvalKit (Duan et al., 2024) to evaluate these benchmarks, and the following is an introduction:

Comprehensive Evaluation: ❶ **MME** (Fu et al., 2025) provides a holistic evaluation of LMMs' perception and cognition across 14 tasks. Its key feature is the use of carefully crafted instruction-answer pairs, which facilitates a straightforward assessment without the need for specialized prompt engineering. ❷ **MMBench** (Liu et al., 2024b) is a cross-lingual benchmark for comprehensively evaluating LMMs. It features over 3,000 bilingual multiple-choice questions spanning 20 skill dimensions, from visual recognition to abstract reasoning.

Optical Character Recognition: ❶ **SEEDBench2_Plus** (Li et al., 2024) benchmarks LMMs on interpreting text-rich visuals (*e.g.,* charts, web layouts). It uses 2,300 multiple-choice questions to test reasoning capabilities where integrating textual and visual information is essential. ❷ **OCRVQA** (Mishra et al., 2019) is a benchmark for evaluating a model's ability to answer questions by reading text within images. It focuses on tasks where textual information is essential, requiring tight integration of visual perception and OCR.

Multidisciplinary Reasoning: ❶ **ScienceQA** (Lu et al., 2022) evaluates scientific reasoning through a large-scale multi-modal benchmark; it features curriculum-based questions with diagrams and provides lectures and explanations for each question to encourage complex reasoning. ❷ **MMMU** (Yue et al., 2023) evaluates LMMs on college-level, multimodal questions requiring expert knowledge. The benchmark includes 11,500 questions from six disciplines, utilizing 30 image formats to test complex, subject-specific reasoning.

Instruction Following: ❶ **MIA-Bench** (Qian et al., 2025) is a targeted benchmark that measures how precisely LMMs can follow complex and multi-layered instructions. It consists of 400 distinct image-prompt combinations engineered to test a model's ability to comply with detailed and nuanced directives.

Multi-Turn Multi Image Dialog Understanding: ❶ **MMDU** (Liu et al., 2025b) evaluates LMMs in multi-image, multi-turn conversational scenarios. It specifically assesses a model's capacity for contextual understanding, temporal reasoning, and maintaining coherence throughout extended interactions.

Mathematical Reasoning: ❶ **MathVista** (Lu et al., 2024) benchmarks the mathematical reasoning of foundation models in visual contexts. It aggregates 6,141 problems from 31 datasets, requiring detailed visual analysis and compositional logic for solution. ❷ **MathVision** (Wang et al., 2024) provides a challenging dataset of 3,040 visually-presented problems from math competitions. Categorized into 16 mathematical areas and five difficulty tiers, it offers a structured evaluation of advanced reasoning in LMMs.

Hallucination: ❶ **HallusionBench** (Guan et al., 2024) diagnoses hallucination and illusion in LMMs' visual interpretations. It employs 346 images and 1,129 structured questions to quantitatively analyze the causes of inaccurate or inconsistent model responses. ❷ **POPE** (Li et al., 2023) evaluates object hallucination in LMMs—the tendency to describe non-existent objects. It uses a polling-based questioning strategy to reliably measure this tendency.

B.3. Evaluation Protocol

To evaluate performance on open-domain question answering tasks, two key metrics are employed: **Cover Exact Match (CEM)** and **F1-Score (F1)**. The **CEM** metric determines whether the ground truth answer is fully contained within the model's prediction (Xu et al., 2025; Jiang et al., 2025; Jia et al., 2026; Du et al., 2025). It is defined by the equation:

$$CEM = \begin{cases} 1, & y_q \subseteq \hat{Y} \\ 0, & \text{otherwise} \end{cases}$$

where y_q represents the ground truth answer and $\hat{Y}$ is the text generated by the model.

The **F1-Score**, on the other hand, assesses the word-level overlap between the predicted and ground truth answers, providing a harmonic mean of **Precision** and **Recall** (Chan et al., 2024; Peng et al., 2025; Fu et al., 2026). Given the ground truth as a set of words $\mathcal{W}(y_q) = \{y_1, \ldots, y_m\}$ and the model's prediction as $\mathcal{W}(\hat{Y}) = \{\hat{y}_1, \ldots, \hat{y}_n\}$, the number of common words is calculated as $\mathcal{U}(\hat{Y}, y_q) = \sum_{t \in \mathcal{W}(y_q)} \mathbf{1}[t \in \mathcal{W}(\hat{Y})]$, where $\mathbf{1}[\cdot]$ is the indicator function.

Based on this, Precision is defined as the fraction of predicted words that are correct, $\mathcal{P}(\hat{Y}, Y) = \frac{\mathcal{U}(\hat{Y}, y_q)}{|\mathcal{W}(\hat{Y})|}$, while Recall represents the fraction of ground truth words that were successfully predicted, $\mathcal{R}(\hat{Y}, Y) = \frac{\mathcal{U}(\hat{Y}, y_q)}{|\mathcal{W}(Y)|}$.

B.4. Baseline Methods

In this section, we provide a brief introduction to the baseline method, as follows:

EWC: This seminal continual learning work (Kirkpatrick et al., 2017) introduces Elastic Weight Consolidation (EWC) to mitigate catastrophic forgetting. EWC slows updates to parameters important for prior tasks by imposing a quadratic constraint based on the Fisher Information Matrix, elastically preserving old knowledge during new learning.

LwF: This work proposes Knowledge Distillation to mitigate catastrophic forgetting (Li & Hoiem, 2017b). The method preserves knowledge by ensuring the new model's predictions on new data align with the old model's outputs, achieving data-free continual learning through output consistency.

LoRA: This highly efficient method, LoRA (Hu et al., 2022), fine-tunes models by training only small, injected low-rank matrices while keeping the original weights frozen. This approach reduces computational costs significantly and helps mitigate catastrophic forgetting.

OLoRA: This work proposes an orthogonal subspace-based method for continual learning (Wang et al., 2023). It allocates independent, orthogonal parameter subspaces for each task, constraining updates to prevent interference and mitigate catastrophic forgetting via an elegant geometric solution.

MoELoRA: The method in (Luo et al., 2024) combines MoE with contrastive learning for PEFT. It specializes experts for different data types and uses contrastive objectives to guide expert collaboration, achieving parameter-efficient fine-tuning that reduces catastrophic forgetting.

SEFE: The method in (Chen et al., 2025) tackles multimodal catastrophic forgetting by separately addressing two types: superficial forgetting of style and essential forgetting of knowledge. A tailored training strategy preserves essential knowledge during continual instruction learning.

CIA: The method (Qiao et al., 2025) addresses the stability-plasticity trade-off by introducing an automated balancing coefficient derived from Taylor expansion of the loss function. This mechanism dynamically weighs gradients and parameters to minimize knowledge interference, while a semantic-driven parameter allocation strategy decides whether to reuse or expand parameters based on instruction similarity.

The implementation of the above baseline methods refer to its open source code or integrated code repository (Chen et al., 2024; Guo et al., 2025).

B.5. Training Parameters about KORE

We have displayed some training parameter settings, as shown in Table 6.

Table 6. Hyperparameter settings for the model training on LLaVA-v1.5 (7B), LLaVA-v1.5 (13B) and Qwen2.5-VL (7B).

LLaVA-v1.5 (7B)				
Rank	**Optimizer**	**Deepspeed**	**Epochs**	**Vision Select Layer**
235	AdamW	Zero3	6	-2
Weight Decay	**Warmup Ratio**	**LR Schedule**	**Learning Rate**	**Batch Size**
0	0.03	cosine decay	2×10^{-4}	54
LLaVA-v1.5 (13B)				
Rank	**Optimizer**	**Deepspeed**	**Epochs**	**Vision Select Layer**
235	AdamW	Zero3	6	-2
Weight Decay	**Warmup Ratio**	**LR Schedule**	**Learning Rate**	**Batch Size**
0	0.03	cosine decay	2×10^{-4}	32
Qwen2.5-VL (7B)				
Rank	**Optimizer**	**Deepspeed**	**Epochs**	**Image Max Pixels**
274	AdamW	Zero3	6	262144
Grad Accum Steps	**Warmup Ratio**	**LR Schedule**	**Learning Rate**	**Batch Size**
8	0.1	cosine decay	2×10^{-4}	24

B.6. Experiment Resources about KORE

All training experiments were conducted using 4 NVIDIA H100 GPUs (each with 96 GiB memory). All evaluation experiments were performed on systems equipped with 4 NVIDIA A100 PCIe GPUs (each with 40 GiB memory).

C. Proof of KORE

In Section § 3.2, KORE-CONSTRAINT initializes the LoRA's low-rank matrix A within the null space of the covariance matrix C, which represents prior knowledge and capabilities. This claim is the premise for KORE-CONSTRAINT's effectiveness, which we formally prove in Theorem C.1. In KORE, the LoRA's low-rank matrix A is frozen, and only matrix B is fine-tuned during the process. We demonstrate in Theorem C.2 why this operation minimizes interference with prior knowledge and capabilities during fine-tuning.

Theorem C.2 extends Theorem C.1: as long as Theorem C.1 ensures that matrix A lies in the null space of the covariance matrix C, the final output of each layer, W^*X, remains approximately equal to W_0X, regardless of how the parameters of matrix B are adjusted.

Theorem C.1. *Let U_{null}^T, W_0, A be the approximate null space of the model's covariance matrix composed of input activations in linear layers, the pre-training weights of the model, and the low rank matrix in LoRA, respectively.*

Proof. We aim to prove that under the assumption that W_0 is full-rank, the column space of A forms a subset of the column space of U_{null}^T, which means $\text{Col}(A) = \text{Col}(U_{\text{null}}^T)$.

Step 1: Based on the definition in Section § 3.2:

$$A = \sqrt{\Sigma^*}(V^*)^T \tag{5}$$

Since Σ^* is a diagonal matrix containing singular values, it only scales the columns of $(V^*)^T$ without changing their span. Therefore, the column space of matrix A is identical to the column space of $(V^*)^T$:

$$\text{Col}(A) = \text{Col}((V^*)^T) \tag{6}$$

Step 2: Based on the SVD of $W_0U_{\text{null}}U_{\text{null}}^T$ in Section § 3.2:

$$W_0 U_{\text{null}} U_{\text{null}}^T = U^* \Sigma^* (V^*)^T \tag{7}$$

V^* represents the right singular vectors of $W_0 U_{\text{null}} U_{\text{null}}^T$ and spans its row space. Since U_{null} is orthogonal, $U_{\text{null}} U_{\text{null}}^T$ projects any matrix onto the subspace spanned by U_{null}. Therefore, when W_0 is full-rank, the column space of $W_0 U_{\text{null}} U_{\text{null}}^T$ is identical to the column space of U_{null}^T:

$$\text{Col}(V^*) = \text{Col}(W_0 U_{\text{null}} U_{\text{null}}^T) = \text{Col}(U_{\text{null}}^T) \tag{8}$$

Step 3: Combining the content of steps 1 and 2:

$$\text{Col}(A) = \text{Col}(V^*) = \text{Col}(U_{\text{null}}^T) \tag{9}$$

Thus, the column space of A is identical to the column space of U_{null}^T, completing the proof.

Theorem C.2. *Let $X^{(l)}$, $W_0^{(l)}$, and $W^{*^{(l)}}$ denote the input activations from pre-trained knowledge, the initial weight matrix of the l-th layer before fine-tuning, and the weight matrix of the l-th layer after fine-tuning, respectively, for the l-th layer of the LMM.*

Proof. We aim to prove that the output of the l-th layer remains approximately unchanged after fine-tuning with KORE, *e.g.,*

$$W^{*^{(l)}} X^{(l)} \approx W_0^{(l)} X^{(l)}, \tag{10}$$

In KORE, the fine-tuned output at the l-th layer is defined as:

$$W^{*(l)} = W_0^{(l)} - B^{(l)} A^{(l)} + B^{*(l)} A^{(l)}. \tag{11}$$

Based on $A^{(l)} X^{(l)} \approx \mathbf{0}$ from Section § 3.2, we have:

$$W^{*(l)} X^{(l)} \approx W_0^{(l)} X^{(l)}. \tag{12}$$

The output thus remains approximately unchanged, ensuring that the fine-tuning minimally alters the pre-trained knowledge. This concludes the proof.

D. More details about analysis of ability to capture knowledge

D.1. Detailed experimental results for capture knowledge

Table 7. The detailed numbers and more results of the experiment in Figure 4

Test Data	Method	Discarded Ranks				
		128	256	512	1024	1536
MME	Plain SVD	1492.95	1487.28	1318.18	1169.87	744.03
	ASVD (with 256 MME samples)	1490.14	1476.02	1488.48	1425.41	1239.74
	CO-SVD (with 256 MME samples)	1498.17	1511.25	1514.43	1486.81	**1458.36**
	CO-SVD (with 32 MME samples)	1508.90	1512.90	1507.78	1498.81	1341.82
	CO-SVD (with 512 MME samples)	1507.42	1516.68	1505.33	1460.32	1449.82
	CO-SVD (with 256 ScienceQA samples)	1486.51	1492.65	1478.73	1419.61	1300.89
ScienceQA	Plain SVD	67.13	66.85	65.59	50.41	0.73
	ASVD (with 256 ScienceQA samples)	67.63	66.95	66.75	62.38	49.14
	CO-SVD (with 256 ScienceQA samples)	67.19	67.16	67.62	67.61	**66.76**
	CO-SVD (with 32 ScienceQA samples)	67.48	66.77	66.97	66.61	64.58
	CO-SVD (with 512 ScienceQA samples)	67.08	67.00	67.40	66.91	66.27
	CO-SVD (with 256 MME samples)	67.74	67.49	67.53	65.69	62.43

Table 7 presents detailed data and additional results from the experiment illustrated in Figure 4. The results indicate that the number of sampled data points has only a limited influence. When the smallest 1536 ranks are discarded, performance with 512 samples is slightly lower than with 256 samples; using 32 samples leads to a more noticeable decline compared to 256 samples, yet still significantly outperforms both Plain SVD and ASVD (Yuan et al., 2023). This suggests that even a small number of samples is sufficient to capture essential knowledge into the covariance matrix.

Furthermore, using test-specific samples allows for better performance after discarding a large number of ranks. For instance, when discarding 1536 ranks, CO-SVD (with 256 MME samples) outperforms CO-SVD (with 256 ScienceQA samples) on the MME, while CO-SVD (with 256 ScienceQA samples) surpasses CO-SVD (with 256 MME samples) on ScienceQA. This demonstrates that CO-SVD effectively captures dataset-specific knowledge and preserves structural features in the covariance matrix, enabling knowledge-oriented constraints and resulting in powerful retention.

D.2. Covariance Visualization Results

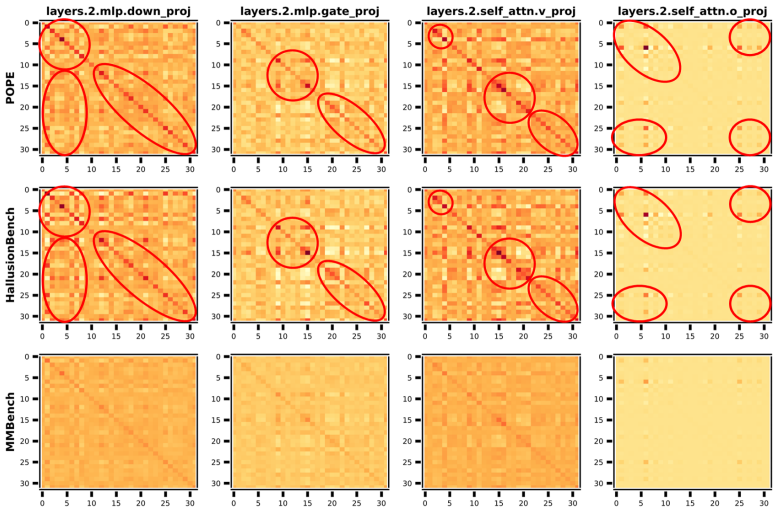

Figure 8. Covariance matrix visualization for “mlp.down_proj”, “mlp.gate_proj”,“self_attn.v_proj”and “self_attn.o_proj” weights in the **2-th layer** on **POPE, HallusionBench and MMBench**.

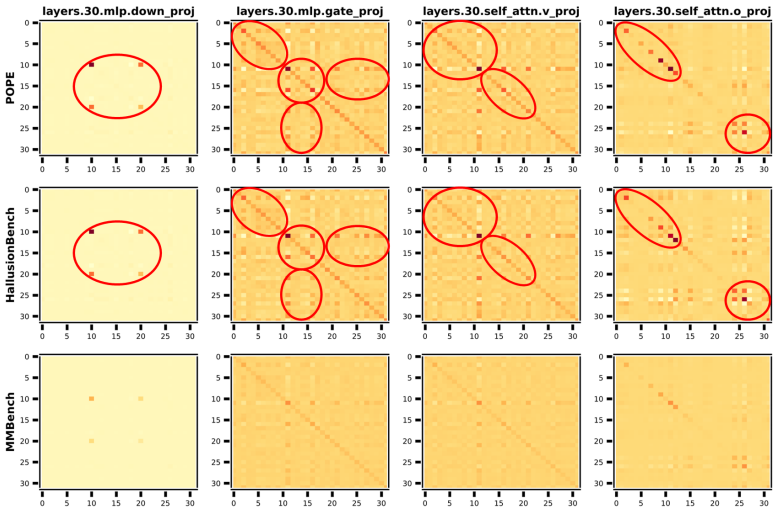

Figure 9. Covariance matrix visualization for “mlp.down_proj”, “mlp.gate_proj”,“self_attn.v_proj”and “self_attn.o_proj” weights in the **30-th layer** on **POPE, HallusionBench and MMBench**.

In Figures 8 and 9, we further provide visualizations of the covariance matrices collected from the POPE, HallusionBench, and MMBench tasks. Due to the high and uninformative original dimensionality of 4096 or 11088, we downsampled the covariance matrices to 32×32 and visualized their heatmaps. We present activations prior to various linear weights, including “mlp.down_proj”, “mlp.gate_proj”,“self_attn.v_proj”and “self_attn.o_proj” from both **layer 2** and **layer 30**. The results

show that heatmaps from **POPE** and **HallusionBench** (both hallucination tasks) share certain similar patterns (highlighted with red circles) not observed in heatmaps from **MMBench**. This indicates that the activated covariance matrices exhibit distinct patterns when inputs from different tasks are processed by the LMMs. These visualizations empirically support that covariance matrix patterns can characterize the triggered task. We leverage such patterns to guide the decomposition of pre-trained weights in LMMs, obtaining initialized adapters enriched with more informative knowledge.

E. More experimental results about KORE

E.1. More Main Results

Regarding the experiment in Figure 5 in § 4.2, we have supplemented Table 8 with detailed numerical performance of all methods on fine-grained knowledge types for readers' reference.

Table 8. **Performance comparison between KORE and baseline methods on fine-grained knowledge types with LLaVA-v1.5 (7B).** PO: Politics; SP: Sports; BU: Business; HE: Health; CE: Celebrity; FI: Film; AL: Album; WR: Written Work.

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑	CEM↑	F1↑
Full-FT	21.35	16.34	12.92	10.99	22.49	20.88	27.31	20.95	19.84	16.47	14.37	13.88	13.11	16.93	12.39	13.16	12.17	7.66	20.34	8.43
LoRA	17.72	19.42	10.54	12.96	19.11	21.50	20.66	24.03	17.81	23.76	12.51	17.09	12.20	21.19	10.57	15.82	10.72	8.72	18.64	12.94
Replay	13.98	19.43	7.61	13.16	15.96	20.69	16.05	22.40	15.38	24.21	8.48	16.39	9.40	18.78	10.34	15.60	3.77	10.79	4.55	8.23
EWC	17.86	21.10	10.45	14.81	19.83	23.02	19.00	24.57	17.41	23.88	12.88	17.58	14.53	22.07	12.16	16.91	10.72	8.13	15.25	17.69
LwF	17.05	21.43	9.62	13.99	19.83	23.66	18.63	25.82	19.03	26.20	11.88	18.40	12.45	21.64	12.39	17.01	9.28	11.11	10.17	17.10
MoELoRA	9.23	14.86	3.39	8.72	6.77	11.77	12.36	18.92	10.53	20.60	3.40	9.28	2.95	10.32	4.43	8.96	3.19	5.22	10.17	14.07
O-LoRA	9.21	14.68	3.67	8.52	7.01	12.23	12.55	18.98	11.74	20.68	3.40	9.22	3.10	10.51	4.20	8.28	3.19	5.35	8.47	12.37
SEFE	21.12	26.93	14.85	20.94	22.61	26.60	23.43	31.88	20.65	33.27	10.84	21.05	13.87	25.58	8.30	14.40	6.67	17.41	15.25	26.27
CIA	17.37	21.71	10.27	14.85	20.56	23.53	18.08	25.46	17.41	25.39	11.35	18.70	12.45	23.21	11.59	17.15	9.57	8.72	11.86	11.72
KORE	**34.74**	**42.96**	**23.83**	**32.31**	**46.19**	**50.38**	**34.69**	**45.74**	**33.20**	**45.23**	**26.17**	**39.39**	**27.79**	**42.61**	**26.93**	**34.05**	**16.52**	**29.54**	**28.81**	**43.05**

E.2. More Results on LMM Scales and Architectures

Regarding the experiment in § 4.3, we supplement the detailed results of knowledge injection in Tables 9, 10 and 11.

Table 9. Performance comparison between KORE and baseline methods with LLaVA-v1.5 (13B) and Qwen2.5-VL (7B).

Method	EVOKE		COM↑	OCR↑	M-DIS↑	INS↑	M-IDU↑	MAT↑	HAL↑	K.A↑	K.R↑	Avg↑	HARS↑
	CEM↑	F1↑											
LLaVA-v1.5 (13B)													
Vanilla	5.33	10.07	66.86	51.12	52.70	66.04	33.93	19.64	56.77	7.70	49.51	28.60	—
LoRA	16.26	22.83	45.28	32.58	43.72	23.26	17.43	15.82	38.08	19.55	32.64	26.09	63.15
Replay	12.05	20.21	**61.65**	**47.51**	48.42	61.04	24.62	19.55	**54.16**	16.13	**45.69**	30.91	66.73
KORE	**32.89**	**44.47**	59.35	45.96	**51.39**	**65.10**	**26.84**	**20.31**	40.52	**38.68**	45.35	**42.01**	**85.46**
Qwen2.5-VL (7B)													
Vanilla	9.34	15.33	81.18	70.32	65.35	78.46	61.25	47.69	66.96	12.34	66.89	39.61	—
LoRA	14.56	14.01	52.54	64.54	22.35	21.39	23.25	13.52	41.38	14.29	38.16	26.22	22.03
Replay	11.73	18.51	**78.54**	**69.17**	65.26	70.20	**50.72**	42.74	**67.48**	15.12	**63.94**	39.53	30.89
KORE	**22.91**	**31.36**	56.60	67.74	**65.48**	**70.51**	45.02	**43.72**	58.57	**27.14**	58.31	**42.72**	**67.10**

Obs 1 in § E.2: KORE still achieves superior knowledge retention performance on larger-scale LMM and different model architectures. As shown in Table 10, on LLaVA-v1.5 (13B), KORE outperforms Replay on seven benchmarks and achieves comparable overall performance. This result demonstrates KORE's potential for superior performance on larger-scale LMM. On Qwen2.5-VL (7B), KORE surpasses LoRA by 20.15 in overall performance, demonstrating its ability to maintain superior knowledge retention across different model architectures and confirming its universality and robustness.

Obs 2 in § E.2: KORE achieves comprehensive performance advantages across diverse knowledge types on both larger-scale LMM and different model architectures. In Table 11, KORE achieves the best knowledge adaptation

Table 10. Performance comparison between KORE and baseline methods on fine-grained knowledge retention evaluations with LLaVA-v1.5 (13B) and Qwen2.5-VL (7B).

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^B ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	MMMU ↑	MIA^B ↑	MMDU ↑	$Math^T$ ↑	$Math^I$ ↑	POPE ↑	$Hall^B$ ↑	
LLaVA-v1.5 (13B)													
Vanilla	65.33	68.38	42.25	59.99	73.90	31.50	66.04	33.93	27.40	11.88	87.07	26.46	49.51
LoRA	30.00	60.57	36.93	28.22	69.13	18.30	23.26	17.43	23.90	7.73	71.64	4.52	32.64
Replay	**57.49**	**65.81**	40.27	**54.75**	70.94	25.90	61.04	24.62	27.00	12.11	**87.09**	**21.23**	**45.69**
KORE	55.99	62.71	**40.32**	51.60	**71.97**	**30.80**	**65.10**	**26.84**	**27.30**	**13.32**	79.29	18.91	45.35
Qwen2.5-VL (7B)													
Vanilla	82.54	79.81	69.61	71.03	72.10	58.60	78.46	61.25	69.70	25.69	86.51	47.42	66.89
LoRA	67.88	37.20	59.29	69.79	42.30	2.40	21.39	23.25	39.40	13.52	73.73	9.02	38.16
Replay	**75.38**	**81.70**	**69.16**	**69.17**	85.12	**45.40**	70.20	**50.72**	**63.90**	21.58	**87.49**	**47.48**	**63.94**
KORE	36.23	76.98	66.80	68.69	**85.55**	**45.40**	**70.51**	45.02	63.10	**24.34**	75.24	41.89	58.31

Table 11. Performance comparison between KORE and baseline methods on fine-grained knowledge types with LLaVA-v1.5 (13B) and Qwen2.5-VL (7B).

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑
LLaVA-v1.5 (13B)																				
LoRA	20.15	25.10	12.65	16.17	24.79	28.69	21.77	29.51	22.27	29.09	11.99	20.34	13.72	25.26	13.18	18.04	6.67	12.18	10.17	15.87
Replay	15.04	21.83	8.16	14.41	15.60	21.76	15.87	24.74	18.62	28.74	8.77	18.42	9.45	21.50	10.91	17.16	5.51	13.38	10.17	20.97
KORE	**36.77**	**46.11**	**25.39**	**34.41**	**47.16**	**53.39**	**37.45**	**50.95**	**35.22**	**48.51**	**28.64**	**42.67**	**28.66**	**44.95**	**31.02**	**38.21**	**22.61**	**35.43**	**20.34**	**33.06**
Qwen2.5-VL (7B)																				
LoRA	17.76	14.09	12.01	7.18	17.41	17.65	22.32	17.90	19.03	17.21	11.06	13.93	8.03	15.91	21.48	14.91	8.70	10.87	16.95	11.32
Replay	13.45	18.40	7.33	11.09	14.03	17.94	14.58	22.72	15.38	23.72	9.84	18.63	7.16	17.69	20.45	28.00	9.28	12.97	16.95	24.89
KORE	**26.93**	**32.51**	**17.42**	**22.75**	**31.20**	**35.11**	**31.00**	**39.43**	**33.20**	**40.49**	**18.51**	**30.11**	**16.11**	**28.63**	**26.14**	**33.20**	**13.33**	**25.91**	**25.42**	**41.24**

performance across all news and entity types on both LLaVA-v1.5 (13B) and Qwen2.5-VL (7B), significantly outperforming LoRA and Replay. This demonstrates that KORE's effectiveness in new knowledge injection is not constrained by model scale or architecture, highlighting its powerful universality.

E.3. More Results on Specific Knowledge-Oriented Constraints

For the experiment on specific knowledge-oriented constraints in § 4.2, we have provided detailed results and presented them below.

Table 12. Performance comparison of specific knowledge-oriented constraints with LLaVA-v1.5 (7B).

Method	EVOKE		COM ↑	OCR ↑	M-DIS ↑	INS ↑	M-IDU ↑	MAT ↑	HAL ↑	K.A ↑	K.R ↑	Avg ↑	HARS ↑
	CEM ↑	F1↑											
KORE	**30.65**	**41.26**	52.41	40.98	48.68	38.54	16.58	18.59	51.75	**35.96**	40.00	37.98	82.81
$KORE_{MME}$	29.48	39.44	**56.90**	39.86	47.41	57.06	**27.70**	17.92	52.20	34.46	**43.03**	38.75	**85.24**
$KORE_{OCR^{VQA}}$	29.95	39.75	52.60	**41.47**	**48.86**	60.10	27.09	18.28	50.15	34.85	42.24	38.54	84.64
$KORE_{Math^T}$	30.06	40.33	52.40	40.32	48.57	60.30	27.69	**19.24**	51.57	35.20	42.68	**38.94**	85.16
$KORE_{Hall^B}$	29.93	39.98	54.37	36.68	46.50	**60.71**	26.30	17.42	**52.67**	34.96	41.86	38.41	84.31

Obs 1 in § E.3: KORE with specific knowledge-oriented constraints achieves superior comprehensive performance. In Table 12, KORE with specific knowledge-oriented constraints (*e.g.,* MME, OCR^{VQA}, $Math^T$, $Hall^B$) causes a slight decrease in knowledge adaptation efficacy, it yields a significant increase in knowledge retention performance on INS and M-IDU, resulting in a superior overall performance.

Obs 2 in § E.3: Specific knowledge-oriented constraints enhance the retention of corresponding knowledge. In Table 13, specific knowledge-oriented constraints enhance the retention of corresponding knowledge without compromising the retention of other knowledge types. This capability underscores KORE's potential for applications requiring customized

knowledge preservation.

Table 13. Performance of specific knowledge-oriented constrains on fine-grained knowledge retention evaluations with LLaVA-v1.5 (7B).

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^B ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	$MMMU^T$ ↑	MIA^B ↑	MMDU ↑	$Math^T$ ↑	$Math^I$ ↑	POPE ↑	$Hall^B$ ↑	
KORE	49.84	54.98	**37.73**	44.24	68.06	29.30	38.54	16.58	25.10	12.09	80.99	22.51	40.00
$KORE_{MME}$	**57.01**	**56.79**	37.51	42.22	66.83	28.00	60.10	**27.70**	24.00	11.84	**81.62**	22.79	**43.03**
$KORE_{OCR^{VQA}}$	50.81	54.38	36.06	**46.88**	**68.22**	29.50	57.06	27.09	24.30	**12.27**	80.82	19.47	42.24
$KORE_{Math^T}$	48.87	55.93	36.41	44.24	67.23	**29.90**	60.30	27.69	**26.50**	11.97	81.04	22.09	42.68
$KORE_{Hall^B}$	55.31	53.44	35.18	38.18	67.30	25.70	**60.71**	26.30	23.10	11.74	80.46	**24.87**	41.86

Table 14. Performance of specific knowledge-oriented constrains on fine-grained knowledge types with LLaVA-v1.5 (7B).

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑
KORE	**34.74**	**42.96**	23.83	**32.31**	**46.19**	50.38	34.69	45.74	33.20	45.23	**26.17**	**39.39**	27.79	**42.61**	**26.93**	**34.05**	**16.52**	**29.54**	28.81	**43.05**
$KORE_{MME}$	34.05	41.53	23.92	31.46	43.17	47.28	34.32	**46.12**	**35.63**	**45.38**	24.48	37.15	27.24	40.96	22.61	30.43	15.07	27.72	**30.51**	42.16
$KORE_{OCR^{VQA}}$	34.46	41.66	**24.29**	31.69	43.53	48.34	**36.35**	46.09	33.20	44.35	25.01	37.65	27.24	41.17	24.09	31.60	14.78	27.16	**30.51**	42.17
$KORE_{Math^T}$	33.71	41.72	22.27	30.39	45.95	**50.88**	33.03	43.38	30.77	43.55	26.06	38.82	**28.15**	42.46	25.80	32.97	15.07	27.37	**30.51**	42.11
$KORE_{Hall^B}$	34.23	41.74	24.11	32.09	43.05	46.98	35.06	44.92	32.39	43.53	25.21	38.05	27.54	41.68	24.66	32.34	14.78	26.86	28.81	40.13

Obs 3 in § E.3: Specific knowledge-oriented constraints also achieve excellent adaptation performance across a wide spectrum of fine-grained knowledge. In Table 14, KORE with specific knowledge-oriented constraints maintains strong adaptation performance across various News and Entity knowledge types, with negligible performance degradation.

E.4. More Results On Ablation Experiments

Regarding the experiment in § 4.4, we have supplemented the experiments in § E.4.1 and § E.4.2.

E.4.1. Rank Ablation Experiments

Table 15. Performance comparison of different ranks with LLaVA-v1.5 (7B).

Method	EVOKE		COM ↑	OCR ↑	M-DIS ↑	INS ↑	M-IDU ↑	MAT ↑	HAL ↑	K.A ↑	K.R ↑	Avg ↑	HARS ↑
	CEM ↑	F1↑											
KORE (rank=64)	24.00	33.07	45.35	29.46	45.02	44.07	19.62	18.08	44.48	28.54	35.70	32.12	75.73
KORE (rank=128)	30.72	40.55	49.97	36.05	47.07	34.87	10.00	17.46	50.30	35.64	37.21	36.42	79.83
KORE (rank=235)	30.65	41.26	52.41	**40.98**	48.68	38.54	16.58	**18.59**	**51.75**	35.96	40.00	37.98	82.81
KORE (rank=256)	**31.05**	**41.32**	**52.48**	39.96	**48.96**	**60.02**	**23.18**	18.09	51.50	**36.19**	**42.10**	**39.14**	**84.93**

Obs 1 in § E.4.1: Increasing the number of trainable parameters enables KORE to achieve stronger performance. In Table 15, KORE's performance in both knowledge adaptation and knowledge retention exhibits a consistent upward trend as the rank and number of trainable parameters increase. This trend is particularly significant on the INS and M-IDU dimensions, which indicates KORE's potential to achieve even stronger performance with larger parameter.

Table 16. Performance of comparison across different ranks on fine-grained knowledge retention evaluations with LLaVA-v1.5 (7B).

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^B ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	$MMMU^T$ ↑	MIA^B ↑	MMDU ↑	$Math^T$ ↑	$Math^I$ ↑	POPE ↑	$Hall^B$ ↑	
KORE (rank=64)	43.63	47.08	33.55	25.36	66.34	23.70	44.07	19.62	**25.20**	10.95	74.22	14.73	35.70
KORE (rank=128)	47.96	51.98	36.32	35.77	67.44	26.70	34.87	10.00	23.90	11.02	79.63	20.97	37.21
KORE (rank=235)	49.84	**54.98**	**37.73**	**44.24**	68.06	29.30	38.54	16.58	25.10	**12.09**	**80.99**	**22.51**	40.00
KORE (rank=256)	**50.06**	54.90	36.89	43.03	**68.51**	**29.40**	**60.02**	**23.18**	24.70	11.48	80.77	22.23	**42.10**

Obs 2 in § E.4.1: Larger trainable parameter scales enhance KORE's knowledge retention performance. In Table 16, KORE (rank=256) achieves near-comprehensive superiority across 12 benchmarks and surpasses KORE (rank=235) by 2.10 in overall performance. This underscores that a larger trainable parameter scale activates stronger knowledge retention in KORE.

Table 17. Performance comparison across different ranks on fine-grained knowledge types with LLaVA-v1.5 (7B).

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑
KORE (rank=64)	28.31	34.84	20.44	27.66	36.64	41.11	28.60	38.13	26.72	35.77	19.27	31.11	21.24	35.25	18.98	25.33	11.01	23.14	22.03	33.44
KORE (rank=128)	34.70	42.07	**24.20**	31.56	44.50	49.17	**36.72**	**47.68**	**34.82**	44.39	26.35	38.89	**28.81**	**43.19**	23.86	30.22	**17.97**	28.77	**35.59**	**44.86**
KORE (rank=235)	34.74	42.96	23.83	**32.31**	**46.19**	**50.38**	34.69	45.74	33.20	**45.23**	26.17	39.39	27.79	42.61	26.93	34.05	16.52	**29.54**	28.81	43.05
KORE (rank=256)	**35.17**	**42.98**	23.92	31.24	45.83	50.35	35.98	47.11	32.79	43.80	**26.55**	**39.49**	28.46	42.74	**27.16**	**34.52**	15.65	26.81	27.12	39.92

Obs 3 in § E.4.1: Larger trainable parameters improve KORE's knowledge adaptation performance on News and Entity types. In Table 17, KORE (rank=256) achieves robust and consistent performance across a broader range of fine-grained knowledge types, demonstrating KORE's potential for superior performance with an increased number of trainable parameters.

E.4.2. SETTING ABLATION EXPERIMENTS

Table 18. Performance comparison of setting ablation with LLaVA-v1.5 (7B).

Method	EVOKE		COM ↑	OCR ↑	M-DIS ↑	INS ↑	M-IDU ↑	MAT ↑	HAL ↑	K.A ↑	K.R ↑	Avg ↑	HARS ↑
	CEM ↑	F1↑											
KORE	30.65	41.26	52.41	**40.98**	**48.68**	**38.54**	**16.58**	18.59	**51.75**	35.96	40.00	**37.98**	**82.81**
W/o Augmentation	10.83	18.31	**59.96**	40.42	47.13	32.53	16.00	**19.71**	49.50	14.57	**40.16**	27.37	64.14
W/o Constraint	**33.93**	**43.71**	46.39	32.38	46.31	32.70	15.38	19.12	46.47	**38.82**	35.78	37.30	79.04
W/o Frozen Matrix A	31.97	41.72	50.73	39.56	48.37	35.30	16.44	19.07	49.91	36.85	38.92	37.88	81.96

Table 19. Performance comparison of setting ablation in knowledge retention with LLaVA-v1.5 (7B).

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^B ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	$MMMU^T$ ↑	MIA^B ↑	MMDU ↑	$Math^T$ ↑	$Math^I$ ↑	POPE ↑	$Hall^B$ ↑	
KORE	49.84	54.98	**37.73**	**44.24**	68.06	**29.30**	**38.54**	**16.58**	25.10	**12.09**	80.99	**22.51**	**51.75**
W/o Augmentation	**58.75**	**61.17**	36.80	44.04	**68.15**	26.10	32.53	16.00	**28.00**	11.41	**81.29**	17.71	40.16
W/o Constraint	40.55	52.23	31.75	33.01	65.81	26.80	32.70	15.38	26.50	11.74	79.16	13.77	35.78
W/o Frozen Matrix A	47.24	54.21	36.01	43.10	67.63	29.10	35.30	16.44	26.70	11.45	80.84	18.98	38.92

Obs 1 in § E.4.2: Modifying KORE's design leads to a degradation in overall knowledge retention performance. In Table 19, the ablated versions W/o Augmentation, W/o Constraint, and W/o Frozen Matrix A exhibit overall performance degradations of 11.59, 15.97, and 12.83 respectively compared to KORE. This significant degradation underscores the high efficacy of KORE's design.

Table 20. Performance comparison of setting ablation on fine-grained knowledge types with LLaVA-v1.5 (7B).

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑
KORE	34.74	42.96	23.83	32.31	46.19	50.38	34.69	45.74	33.20	45.23	26.17	39.39	27.79	42.61	26.93	34.05	16.52	29.54	**28.81**	**43.05**
W/o Augmentation	14.04	20.22	8.25	14.06	15.96	20.08	14.39	23.13	14.57	25.69	7.30	16.21	8.08	20.13	8.41	14.15	3.77	6.56	13.56	22.27
W/o Constraint	**38.45**	**45.16**	**25.57**	**32.56**	**46.43**	**50.66**	**41.33**	**51.22**	**36.84**	45.78	**28.97**	**42.12**	**29.67**	**44.18**	**30.45**	**37.75**	**20.58**	**33.59**	**28.81**	40.02
W/o Frozen Matrix A	36.49	43.42	25.11	31.70	**46.43**	50.44	37.82	48.20	36.44	**46.44**	27.01	39.85	28.05	42.88	27.95	34.57	19.13	30.95	**28.81**	39.86

Obs 2 in § E.4.2: W/o Constraint yields superior knowledge adaptation performance across a wide spectrum of fine-grained knowledge. In Table 20, W/o Constraint achieves superior knowledge adaptation performance on fine-grained News and Entity types. These gains stem from KORE-AUGMENTATION's ability to perform profound and structured augmentation.

E.5. More Results On Comparison With General Augmentation methods

Obs 1 in § E.5: KORE-AUGMENTATION demonstrates absolute comprehensive performance superiority in knowledge retention evaluations. In Table 22, KORE-AUGMENTATION surpasses the best general augmentation method by a margin of 10.76 in overall performance, demonstrating its substantially superior capability for knowledge retention.

Table 21. Performance comparison of different augmentation methods with LLaVA-v1.5 (7B).

Method	EVOKE		COM ↑	OCR ↑	M-DIS ↑	INS ↑	M-IDU ↑	MAT ↑	HAL ↑	K.A ↑	K.R ↑	Avg ↑	HARS ↑
	CEM ↑	F1↑											
KORE-AUGMENTATION	**33.93**	**43.71**	46.39	**32.38**	**46.31**	32.70	15.38	19.12	**46.47**	**38.82**	**35.78**	**37.30**	**79.04**
Augmentation for Text													
Knowledge-Agnostic	12.10	19.10	**53.50**	25.38	39.79	**44.37**	**22.41**	18.47	43.72	15.60	35.71	25.65	63.54
Knowledge-Aware	16.94	23.64	48.85	28.10	40.28	43.59	20.72	18.14	41.65	20.29	34.86	27.58	69.43
Augmentation for Images													
Knowledge-Agnostic	15.46	21.20	51.42	13.22	36.94	43.65	21.60	17.09	41.27	18.33	32.09	25.21	64.71
Knowledge-Aware	14.68	21.97	51.40	24.57	40.54	40.84	18.53	**19.23**	38.70	18.33	34.02	26.17	66.48

Table 22. Performance comparison of different augmentation methods in knowledge retention with LLaVA-v1.5 (7B).

Method	COM		OCR		M-DIS		INS	M-IDU	MAT		HAL		Avg
	MME ↑	MM^{B} ↑	$SEED^{B2P}$ ↑	OCR^{VQA} ↑	SQA ↑	$MMMU^{T}$ ↑	MIA^{B} ↑	MMDU ↑	$Math^{T}$ ↑	$Math^{I}$ ↑	POPE ↑	$Hall^{B}$ ↑	
KORE-AUGMENTATION	40.55	52.23	**31.75**	**33.01**	65.81	**26.80**	32.70	15.38	**26.50**	11.74	**79.16**	**13.77**	**46.47**
Augmentation for Text													
Knowledge-Agnostic	**51.67**	**55.33**	25.99	24.77	64.38	15.20	**44.37**	**22.41**	25.20	11.74	79.04	8.40	35.71
Knowledge-Aware	50.02	47.68	24.95	31.25	65.75	14.80	43.59	20.72	24.20	12.07	74.05	9.24	34.86
Augmentation for Images													
Knowledge-Agnostic	50.43	52.41	11.86	14.58	64.18	9.70	43.65	21.60	22.60	11.58	73.95	8.58	32.09
Knowledge-Aware	51.35	51.46	27.23	21.91	**66.29**	14.80	40.84	18.53	21.20	**17.26**	69.71	7.68	34.02

Table 23. Performance comparison of different augmentation methods on fine-grained knowledge types with LLaVA-v1.5 (7B).

Method	News										Entity									
	Avg		PO		SP		BU		HE		Avg		CE		FI		AL		WR	
	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑	CEM ↑	F1 ↑
KORE-AUGMENTATION	**38.45**	**45.16**	**25.57**	**32.56**	**46.43**	**50.66**	**41.33**	**51.22**	**36.84**	**45.78**	**28.97**	**42.12**	**29.67**	**44.18**	**30.45**	**37.75**	**20.58**	**33.59**	**28.81**	**40.02**
Augmentation for Text																				
Knowledge-Agnostic	14.59	20.11	8.52	14.84	17.05	21.34	17.16	24.56	14.57	23.34	9.37	17.99	10.52	22.06	6.59	10.74	8.12	15.00	13.56	21.25
Knowledge-Aware	20.19	24.99	11.37	16.28	24.55	28.48	21.96	29.00	19.03	28.94	13.37	22.17	13.92	26.33	12.95	18.16	9.57	12.99	13.56	18.90
Augmentation for Images																				
Knowledge-Agnostic	18.38	22.42	10.72	14.88	22.97	26.92	20.11	26.60	19.84	27.28	12.26	19.87	12.35	23.27	13.07	16.59	10.43	16.13	15.25	14.83
Knowledge-Aware	17.15	23.01	9.99	14.93	19.35	24.35	18.08	25.79	16.19	27.05	11.97	20.84	12.86	24.29	13.86	19.59	7.25	11.47	15.25	21.78

Obs 2 in § E.5: KORE-AUGMENTATION demonstrates superior knowledge adaptation performance across a wide spectrum of fine-grained knowledge types. In Table 23, KORE-AUGMENTATION achieves the best performance on all all knowledge types, demonstrating its superiority over general augmentation methods for new knowledge injection.

F. Convergence comparison of various methods via loss curves.

Figure 10 presents the training loss curves of the six methods, providing an intuitive comparison of their convergence behaviors. Although KORE and the baseline methods use different training datasets, the loss curves reveal that O-LoRA and SEFE fail to fit the EVOKE's knowledge injection dataset. While LoRA, EWC, and Full-FT converge to very low loss values and successfully fit the evoke dataset, their performance in Table 1 indicates poor generalization to new knowledge, suggesting overfitting. In contrast, KORE not only converges effectively on the KORE-74K dataset but also demonstrates strong generalization capabilities for novel knowledge.

G. More details about KORE-AUGMENTATION

G.1. More construction process about KORE-AUGMENTATION

In this section, we elaborate on the implementation of KORE-AUGMENTATION. The fully automated construction pipeline and a data example are illustrated in Figure 11. The following details each step of the pipeline:

Step 1: Constructing Multi-rounds of Dialogue. We design strict rules and diverse task examples, employing GPT-4o to generate multi-turn dialogue data based on the original knowledge. The first turn is a heuristic QA pair randomly selected from templates, such as:

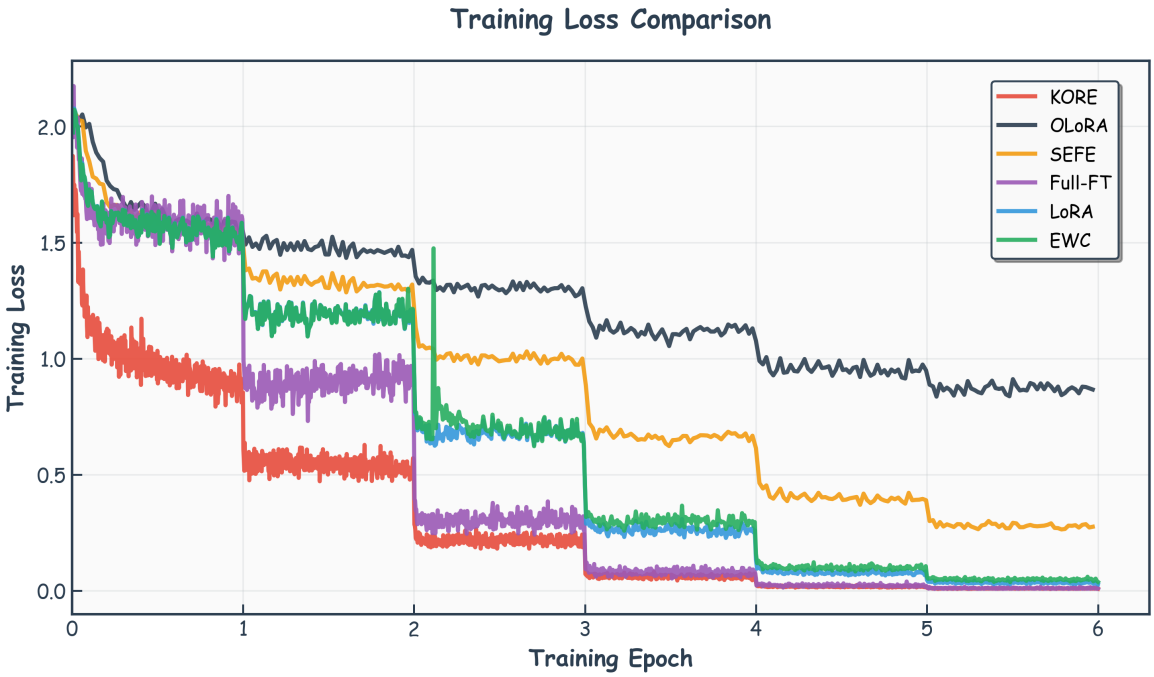

Figure 10. **The training loss curves on EVOKE of Full-FT, LoRA, EWC, O-LoRA, SEFE and KORE.** It should be clarified that Full-FT, LoRA, EWC, O-LoRA, and SEFE are trained using the knowledge injection dataset from EVOKE, whereas KORE is trained using the KORE-74K dataset. The scale of the training data differs between these setups, resulting in varying numbers of iteration steps per epoch. Consequently, KORE exhibits a rapid decrease in loss during the first epoch. The purpose of reporting this loss graph is to provide readers with an intuitive understanding of the convergence of various methods.

<''Please explain the {type} news that is shown in the image.'', ``The image provides the following {type} news summary: {title}.''>

<''Please tell me what the {type} entity in this image is.'', ``The {type} entity shown in the picture is {entity_name}.''>

The remaining dialogue data are generated automatically by GPT-4o. For each instance, we first generate up to 10 dialogue questions based on the original knowledge and predefined rules. Then, the corresponding answers are produced using the original knowledge, the generated questions, and the rules as input. The query images are taken directly from the original image set. This process results in a complete multi-rounds dialogue dataset, obtaining 9,422 rounds of multi-rounds dialogue data and 75,710 rounds of dialogue. Further templates and prompt designs are provided in § I.1.

Step 2: Collecting Recognition and Caption Images. We use news titles or entity names as search keywords to retrieve and download the top five images via the Google search engine. CLIP (Radford et al., 2021) is then employed to extract visual features from both the downloaded and original images. We compute cosine similarity between them and retain the two images with the highest similarity scores, excluding any identical matches (similarity $\neq 1$). These selected images serve as query images for visual recognition and image captioning tasks. **Step 3: Constructing Visual Recognition QA.** For this task, templates are first manually created. Questions are randomly selected from these templates, and the answer is defined as "Yes". The instruction content is "Answer this question with Yes or No.", and the query image is randomly chosen from the images obtained in Step 2. A template example is provided below:

<''Is the image depicting news {title}?''>

<''Can you see {entity_name} in this picture?''>

Further templates and prompt designs are provided in § I.2.

Step 4: Constructing Image Caption QA. We first establish rigorous rules and diverse task examples. Using GPT-4o, we generate summary data based on original knowledge to serve as answers for the image caption task. The instruction content is "Answer this question in one paragraph.', and the query image corresponds to the remaining images from Step 2.

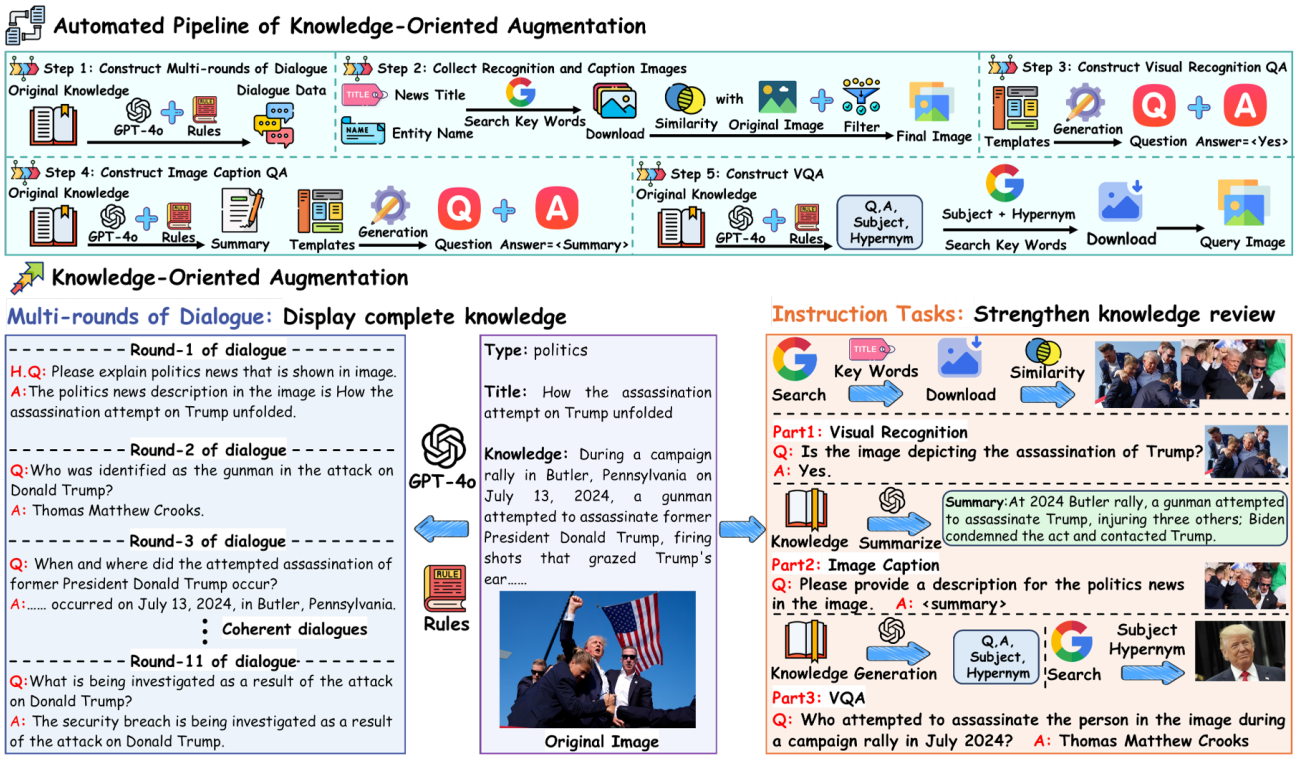

Figure 11. **Overview of construction pipeline for KORE-74K.** The entire data construction process is automated, with only the question templates being manually crafted.

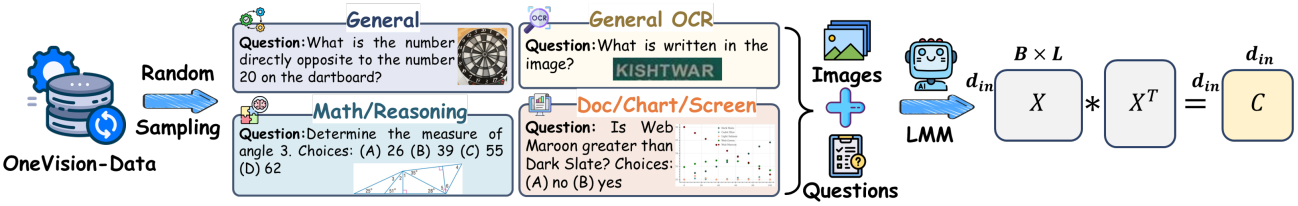

Figure 12. The process of sampling using the OneVision dataset.

Questions are randomly selected from a template, such as:

<``Could you please describe the {type} news shown in the picture?''>

<``Please provide a description for the {type} entity in the image.''>

Further templates and prompt designs are provided in § I.3.

Step 5: Constructing VQA. First, strict rules and diverse task examples are established. Using GPT-4o, quadruplets ⟨Question, Answer, Subject, Hypernym⟩ are generated based on original knowledge, for instance, <"Who attempted to assassinate the person in the image during a campaign rally in July 2024?", "Thomas Matthew Crooks", "Donald John Trump", "Person">. Subsequently, the subject and hypernym are combined as search keywords to retrieve and download the top 1-ranked image from Google, thereby constructing VQA data. Further prompt designs are provided in § I.4.

Through the above automated pipeline, we have augmented the EVOKE's knowledge injection dataset to KORE-74K, which can better achieve knowledge adaptation.

G.2. More statistical analysis about KORE-AUGMENTATION

In Table 24, we provide detailed statistical data analysis of KORE-74K.

Table 24. **Key Statistics of KORE-74K.**

Statistic	Number
Total data	74,734
- Multi-rounds of dialogue data	9,422 (12.6%)
- Visual recognition data	9,422 (12.6%)
- Image caption data	9,422 (12.6%)
- VQA data	46,468 (62.2%)
Number of dialogue rounds	75,710
Number of unique images	65,312
Maximum question length	44
Maximum answer length	143
Average question length	15.5
Average answer length	11.9

G.3. The process of sampling using the OneVision dataset

The sample process for OneVision data is shown in Figure 12.

H. Case Study

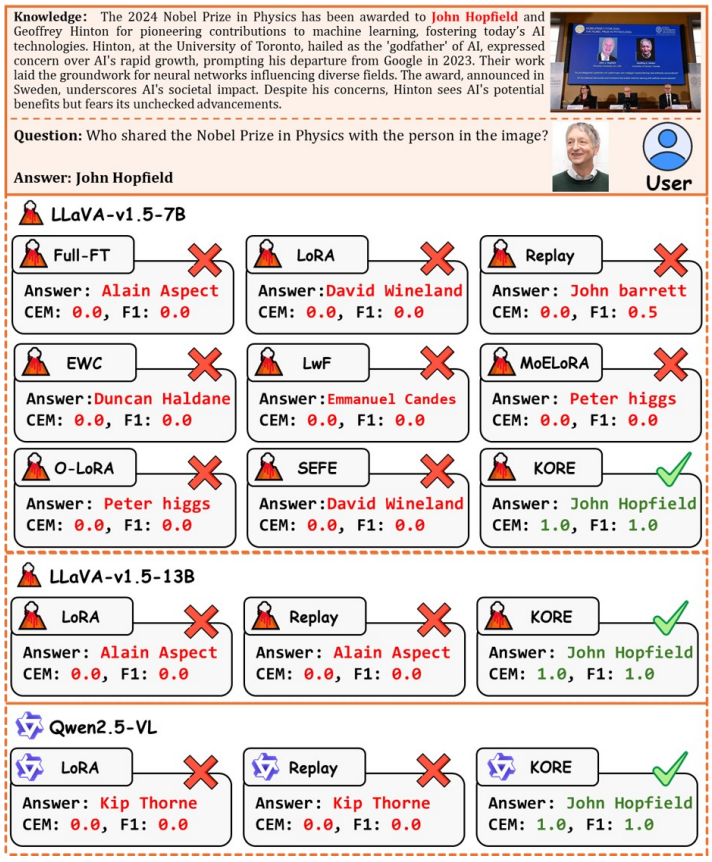

Figure 13. Case Study of News.

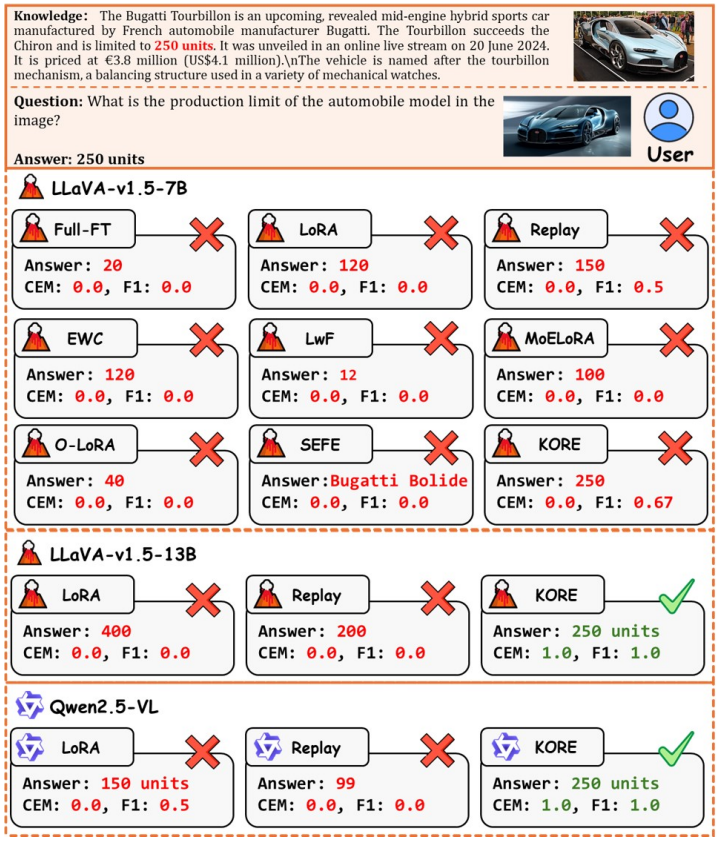

Figure 14. Case Study of Entity.

I. Prompt

I.1. Prompt details regarding Multi-rounds of Dialogue

Prompts and Templates 1 (Part 1): Multi-rounds of Dialogue

Generation Question Prompt:

System Prompt:

"You have received a descriptive text that provides you with the knowledge, events, and definitions described in the text. You need to generate questions coherently and cover as much of the descriptive text as possible. You just need to output the problem. The maximum number of generated questions is 10. If the previously generated questions are sufficient to cover the entire descriptive text, the output questions can be less than 10."

"From the provided descriptive text, create up to 10 coherent questions that comprehensively cover its content. Your output should consist only of the questions. It is acceptable to generate fewer than 10 questions if the material has been fully covered."

"You are required to formulate a set of coherent questions from a given descriptive text, covering its contents as completely as possible. The number of questions must not exceed 10, but it is permissible to output fewer if they adequately cover the text. The sole output should be the questions."

"Generate a series of logical questions that cover all the knowledge, events, and definitions in the descriptive text you have received. While the maximum number of questions is 10, you can output a smaller number if the text is fully addressed. Please ensure you only output the questions."

"Your task is to generate questions based on a descriptive text, ensuring they are coherent and cover its knowledge, events, and definitions as thoroughly as possible. You should generate a maximum of 10 questions and only output the questions themselves. You may provide fewer than 10 if they are sufficient to cover the entire text."

User Prompt:

"News: {news} Please generate questions."
"Given the news: {news} Please generate questions."
"Can you generate questions for the following news: {news}."
"Generate questions for the following news: {news}."
"Please generate questions based on the following news: {news}."

Prompts and Templates 1 (Part 2): Multi-rounds of Dialogue

Generation Answer Prompt:

System Prompt:

"You have gained knowledge and a problem to be solved. You need to answer this question based on the content of your knowledge. Output your answer."

"You now have the necessary knowledge and a specific problem. Based only on this information, provide your answer to the question and output the result."

"You are equipped with the required information and a problem to resolve. Formulate your answer based solely on the content of this knowledge and then output it."

"Using the knowledge you have been given, solve the problem presented. Your response must be based exclusively on this information. Please output your answer."

"Now that you have the relevant knowledge and the question, you must provide a solution. Ensure your answer is derived strictly from the provided content, then output your response."

User Prompt:

"Given the knowledge: {knowledge} Answer the following question: {question}."

"Knowledge: {knowledge} Answer the following question: {question}."

"Answer the following question based on the knowledge: Knowledge:{knowledge} Question: {question}."

"Here is some knowledge: {knowledge} nNow, answer the following question: {question}."

"You are given the knowledge:{knowledge} Can you answer the following question:{question}."

Prompts and Templates 1 (Part 3): Multi-rounds of Dialogue

Heuristic question templates for News:
"What is the {type} news in the image about?"
"Could you summarize the {type} news story presented in the image?"
"What is the {type} news event being depicted in this picture about?"
"Please explain the {type} news that is shown in the image."
"Can you tell me what the {type} news in this image is about?"

Heuristic answer templates for News:
"The {type} news description in the image is {title}."
"The {type} news in the image can be described as {title}."
"According to the image, the {type} news description is {title}."
"The image provides the following {type} news summary: {title}."
"The {type} news content shown in the picture is {title}."

Heuristic answer templates for Entity:
"What is the {type} entity in the image?"
"Can you identify the {type} entity shown in the picture?"
"What is the {type} entity depicted in this image?"
"Please tell me what the {type} entity in this image is."
"What {type} entity is visible in the photo?"

Heuristic answer templates for Entity:
"The {type} entity in the image is {entity_name}."
"The {type} entity shown in the picture is {entity_name}."
"The {type} entity depicted in the image is {entity_name}."
"The {type} entity illustrated in the picture is {entity_name}."
"The {type} entity present in the image is {entity_name}."

I.2. Prompt details regarding Visual Recognition QA

Prompts and Templates 2: Visual Recognition QA

Question templates for News:
"Is the image depicting news {title}? Answer this question with Yes or No."
"Does this image illustrate the news titled {title}? Answer this question with Yes or No."
"Is this picture related to the news with the headline {title}? Answer this question with Yes or No."
"Is the image about the news report named {title}? Answer this question with Yes or No."
"Does this photo correspond to the news {title}? Answer this question with Yes or No."

Question templates for Entity:
"Is {entity_name} in the image? Answer this question with Yes or No."
"Does the image show {entity_name}? Answer this question with Yes or No."
"Can you see {entity_name} in this picture? Answer this question with Yes or No."
"Is {entity_name} visible in the image? Answer this question with Yes or No."
"Does this picture contain {entity_name}? Answer this question with Yes or No."

I.3. Prompt details regarding Image Caption QA

Prompts and Templates 3 (Part 1): Image Caption QA

Question templates for News:
"Please provide a description for the {type} news in the image. Answer this question in one paragraph."
"Could you please describe the {type} news shown in the picture? Answer this question in one paragraph."
"Please offer a description of the {type} news depicted in the image. Answer this question in one paragraph."
"Please give a description of the {type} news depicted here. Answer this question in one paragraph."
"Can you tell me about the {type} news featured in the photograph? Answer this question in one paragraph."

Answer templates for News:
"The image depicts {title}. {summary}"

Question templates for Entity:
"Please provide a description for the {type} entity in the image. Answer this question in one paragraph."
"Could you please describe the {type} entity shown in the picture? Answer this question in one paragraph."
"Please offer a description of the {type} entity depicted in the image. Answer this question in one paragraph."
"Please give a description of the {type} entity depicted here. Answer this question in one paragraph."
"Can you tell me about the {type} entity featured in the photograph? Answer this question in one paragraph."

Answer templates for Entity:
"The image depicts {entity_name}. {summary}"

Prompts and Templates 3 (Part 2): Image Caption QA

Generation Summary Prompt:

System Prompt:

"You have acquired a piece of knowledge, and now you need to condense it into a paragraph of no more than 25 words, while trying to maintain the original meaning of the knowledge as much as possible."

"Your task is to take a piece of knowledge you've learned and summarize it. The summary must be a paragraph of 25 words or less, while retaining the original meaning."

"You need to distill the information you have acquired into a concise paragraph. Ensure it does not exceed 25 words and preserves the essence of the original knowledge as accurately as possible."

"Condense a concept you have just learned into a brief paragraph. You must adhere to a 25-word limit, all while making sure the core message remains intact."

"Take the new information you possess and shorten it into a single paragraph. This condensed version must be under 25 words and should accurately reflect the original meaning."

User Prompt:

"Knowledge: {knowledge} Please summarize this knowledge."

"Given the knowledge: {knowledge} Please summarize this knowledge."

"Can you summarize this content for the following knowledge: {knowledge}."

"Summarize questions for the following knowledge: {knowledge}."

"Please summarize this content based on the following knowledge: {knowledge}."

I.4. Prompt details regarding VQA

Prompts and Templates 4: VQA

Generation Quadruplets Prompt:

System Prompt:

"You have acquired a piece of knowledge and are now required to generate up to 5 questions based on it. For each generated item, you must provide the question itself, its answer (which should be a word or short phrase), a subject object extracted from the question, and that subject's hypernym. When extracting the subject object, you must follow a critical rule: the subject must be a specific entity that is explicitly mentioned within the question itself, serving as a key reference point. Crucially, this extracted subject cannot be the answer to the question. A helpful test for identifying the correct subject is to check if its name could be logically replaced by a placeholder, such as this company or the entity in the image, while the question remains coherent. If the provided knowledge is fully covered by fewer than 5 questions, you may generate fewer."

"Your task is to generate up to five question sets from the provided knowledge. Each set must include the question, a brief answer (word/phrase), a subject object, and its hypernym. When selecting the subject object, you must follow a key rule: it must be a specific entity explicitly named in the question and cannot be the answer. A good test is to see if a placeholder like this entity can logically replace it. Fewer than five questions are fine if the knowledge is fully covered."

"Based on the knowledge you've acquired, create a maximum of five questions. For each, provide a short answer, identify a subject object, and state its hypernym. The 'subject object' must adhere to this critical constraint: it must be a specific entity mentioned directly in the question that serves as a reference point but is not the answer. To verify your choice, check if substituting a generic term like this item would keep the question coherent. You may generate fewer questions if they are sufficient."

"You are required to produce up to five questions from the given information. For each item, output the question, its short answer, a subject object, and that subject's hypernym. The rule for extracting the subject object is that it must be a specific, named entity within the question's text and must be different from the answer itself. A helpful check is to replace its name with a placeholder (*e.g.,* this organization) to see if the question still makes sense. Fewer questions are acceptable if the topic is fully addressed."

"Formulate as many as five questions based on the knowledge. Each output must consist of the question, a concise answer, an extracted subject object, and its hypernym. A crucial guideline applies: the subject object must be a specific entity named in the question that the query revolves around, but it cannot be the answer. You can confirm the correct subject by checking if a placeholder such as the specified object could logically take its place. Generating all five questions is not necessary if the knowledge is completely covered."

User Prompt:

"Knowledge: {knowledge} Please generate questions, answers, subjects, hypernyms."

"Given the knowledge: {knowledge} Please generate questions, answers, subjects, hypernyms."

"Can you generate questions, answers, subjects, hypernyms for the following knowledge: {knowledge}."

"Generate questions, answers, subjects, hypernyms for the following knowledge: {knowledge}."

"Please generate questions, answers, subjects, hypernyms based on the following knowledge: {knowledge}."