GENERATING TO GENERALIZE: SOCRATIC QUESTIONING FOR COMPOSITIONAL CONTINUAL VQA

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual visual question answering (VQA) requires models to retain previously learned knowledge while reasoning compositionally across different tasks, yet existing generative continual learning methods primarily emphasize knowledge retention and overlook cross-task composition. We address this limitation with a Mixture-of-Experts framework comprising dual-purpose experts that learn visual question answering (VQA) and visual question generation (VQG) within a shared multimodal large language model backbone. Each expert is implemented via low-rank adaptation to preserve task-specific knowledge. In contrast, knowledge distilled from prior experts provides complementary views to new experts, enabling compositional generalization without storing past raw experiences. A task-agnostic router selects experts at inference time. To more rigorously evaluate compositional inference in continual VQA, we introduce a human-annotated benchmark comprising real-world compositional questions. Extensive experiments on both standard and our compositional VQA benchmarks demonstrate that our approach retains knowledge and achieves stronger compositional generalization than prior generative continual learning methods.

1 INTRODUCTION

Imagine a child who first learns to name colors, then later learns to count objects. At each stage, the child is expected not only to acquire the new skill but also to retain the previous one. Continual learning (CL) aims to emulate this process by enabling models to acquire knowledge sequentially while preserving previously learned knowledge, thereby avoiding catastrophic forgetting (CF), i.e., the loss of previously acquired knowledge when learning new tasks (French, 1999). Despite substantial progress, most CL research has focused on single-modal classification with closed output spaces, where common strategies such as replay, architectural constraints, or task-specific modules still struggle to balance stability and plasticity (Wang et al., 2024a).

Recently, multimodal models have attracted growing attention for their ability to learn joint representations across modalities. However, they remain susceptible to forgetting, and most multimodal CL studies are still limited to classification settings (Thengane et al., 2022). In parallel, multimodal large language models (MLLMs) have shifted the focus toward reasoning and generation, yielding a richer and more practical testbed for continual learning (Guo et al., 2025). Unlike closed-set classification, generative continual learning must preserve task-specific knowledge, support reasoning, and transfer among tasks. This setting remains largely underexplored (Huai et al., 2025).

Visual Question Answering (VQA) is a representative instance of this challenge. Although VQA can be cast as a classification task, its generative formulation is more natural and scalable. To study continual VQA, Zhang et al. (2023) partitions VQA v2 (Goyal et al., 2017) into tasks defined by linguistically motivated question types, enabling evaluation of incremental reasoning and multimodal retention. Only a few works, notably VQACL (Zhang et al., 2023) and CL-MoE (Huai et al., 2025), have explored generative continual VQA, and their focus has primarily been on knowledge retention.

However, retaining performance on individual tasks is only part of the problem. In realistic settings, learners must also combine previously learned skills. Continuing the earlier example, a child who has separately learned counting (e.g., "How many trucks are there?") and color (e.g., "What color is the flip flop?") should eventually answer questions such as "How many different colors of

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

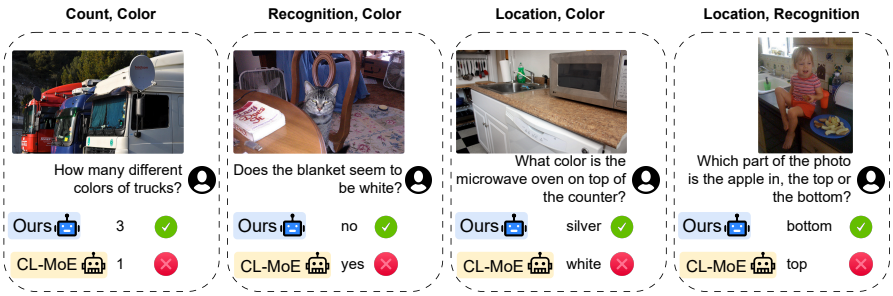


Figure 1: Comparison of Our MoE Design vs. CL-MoE (Huai et al., 2025) on Compositional Questions: When a query requires combining skills from multiple seen tasks (e.g., Count and Color), CL-MoE often fails. Our MoE design explicitly models task composition, enabling more accurate and contextually appropriate responses.

trucks are there?" when shown multiple trucks of different colors. This illustrates a more realistic multimodal setting, in which task boundaries are soft (e.g., count-color task overlap) and previously acquired capabilities must compose naturally at inference time. In this work, we define compositionality specifically as task compositionality, i.e., questions that require jointly applying knowledge acquired from two or more previously learned tasks to produce a correct answer. Yet VQACL (Zhang et al., 2023), constructed from VQA v2 (Goyal et al., 2017), inherits its limited compositional structure (Hudson & Manning, 2019) and does not explicitly construct questions that compose multiple task types, thereby reducing the need for compositional reasoning and steering prior methods toward knowledge retention in continual learning. As a result, they achieve strong per-task accuracy but still fail on simple compositions, showing that preserving task-specific competence does not, by itself, yield compositional reasoning. For example, CL-MoE (Huai et al., 2025) achieves an average performance of 51.34% on individual tasks but drops to 39.60% on compositional questions. We illustrate such failure cases in Fig. 1. A similar trend is observed for other generative CL methods, such as QUAD (Marouf et al., 2025) and GaB (Das et al., 2025) (See Sec. 6).

Although Mixture of Experts (MoEs), such as CL-MoE (Huai et al., 2025), show promise in knowledge retention, the aforementioned gap highlights a central design challenge for MoE in CL. Methods that strongly isolate parameters can better preserve past knowledge, but they make cross-task reuse difficult (Wang et al., 2024a). Conversely, heavily shared parameterizations encourage transfer, yet become increasingly brittle as the number and diversity of tasks grow, often leading to overgeneralized representations and reduced retention (Wang et al., 2024b; Christianos et al., 2021). To overcome this tradeoff, expandable network architectures have shown promise in CL by balancing parameter isolation with controlled knowledge reuse (Rusu et al., 2022). Building on this foundation, more recent adaptive expansion approaches reuse previously learned representations as a knowledge base for subsequent tasks (Ye & Bors, 2025; 2026; Ge et al., 2025). Collectively, these works suggest that access to prior experts is not simply an efficiency cost, but a principled mechanism for knowledge reuse.

Motivated by this perspective, we propose an MoE-based architecture for continual VQA that jointly targets retention and compositionality. Our method isolates parameters across experts to stabilize previously acquired knowledge, while enabling inter-expert communication to support cross-task reasoning. Concretely, each expert is dual-purpose: visual question generation (VQG) serves as an auxiliary task alongside visual question answering (VQA), thereby creating teacher–student interactions in which previously learned experts guide the current one and promote knowledge fusion without overwriting isolated parameter spaces. Specifically, past dual-purpose experts provide Socratic questioning (i.e., complementary perspectives (see App Fig. 4)) on current-task images, encouraging the new expert to reason more openly rather than overfitting to task-specific patterns. Prior experts, therefore, remain reusable sources of multimodal knowledge. At the same time, the associated overhead is kept practical through low-rank adaptation, a lightweight language-based router, and the absence of memory buffers. The resulting architecture simultaneously supports retention through parameter isolation, task-agnostic generalization through routing, and compositional reasoning through Socratic questioning.

To evaluate compositional reasoning directly, we introduce a human-annotated benchmark in which two or more task skills are combined into a single question-answer pair. Our benchmark contains

500 compositional examples spanning 100 compositional question types, each mapped to previously learned tasks. This broad task coverage ensures a representative assessment of the model's ability to compose learned skills, and complements it with an additional public benchmark. Existing compositional QA datasets, such as GQA (Hudson & Manning, 2019) and CLEVR (Johnson et al., 2017), do not correspond to the task boundaries used in continual VQA and therefore fail to capture the incremental, real-world compositions that arise in this setting. Human annotation was thus necessary to ensure both task alignment and coverage of realistic compositional queries. In summary:

- **Retention and compositional reasoning.** We propose an MoE-based architecture with parameter isolation to retain knowledge, together with dual-purpose experts and inter-expert communication to fuse and recombine learned knowledge across different tasks, enabling compositional reasoning.
- **Buffer-free expandable design.** Our strategy avoids storing raw exemplars, reduces parameter overhead using low-rank adaptation, and uses a lightweight language-based router to support task-agnostic generalization.
- **Broader evaluation.** We introduce the first human-annotated benchmark for task compositionality in continual VQA and evaluate our method on it alongside the VQACL benchmark and our GQA-derived COCO-GQA split, demonstrating improved knowledge retention, and stronger compositional reasoning beyond standard per-task evaluation.

2 Related Work

Multimodal learning involves training models on two or more data modalities, such as audio, video, text, or images (Wu et al., 2023). When these models are built on top of large language models (LLMs) (e.g., OPT (Zhang et al., 2022), FlanT5 (Chung et al., 2024)) and process multiple modalities, they are referred to as multimodal large language models (MLLMs) (Alayrac et al., 2022; Bai et al., 2023). Examples such as LLaVA (Liu et al., 2023), and BLIP-2 (Li et al., 2023), demonstrate that models can generalize across diverse vision-language tasks. More recently, research has focused on enhancing the generative and reasoning capabilities of MLLMs (Singh et al., 2026; OpenAI, 2024). Such generalization is particularly important in the continual learning (CL) setting, where models must learn a sequence of tasks without revisiting previously seen data (Wang et al., 2024a). A core challenge in this setting is catastrophic forgetting (CF), where models tend to lose prior knowledge while learning new knowledge (Kirkpatrick et al., 2017). Although MLLMs demonstrate impressive reasoning and generalization, they are not immune to forgetting (Li et al., 2023). Classical CL strategies, including parameter regularization (Kirkpatrick et al., 2017), replay-based methods (Chaudhry et al., 2019; Buzzega et al., 2020; Zhang et al., 2023), and architectural adaptations (Rusu et al., 2022) have been proposed to address those issues, but they are not specifically tailored for MLLMs. Moreover, existing CL benchmarks were not built for MLLMs and do not fully capture their abilities. A practical and representative setting that requires reasoning skills is Visual Question Answering (VQA) (Antol et al., 2015; Whitehead et al., 2021), which assesses understanding of both visual and textual modalities to produce open-ended, reasoning-driven responses. Zhang et al. (2023) introduced a continual VQA for generative answers, offering a more suitable evaluation setting for MLLMs. The authors proposed a memory-based method to tackle the challenges of continual VQA. However, this approach lacks scalability and does not fully prevent CF. Marouf et al. (2025) proposed a question-only replay method that uses attention-consistency distillation, yielding only limited performance gains. Das et al. (2025) proposed a memory-free method based on generative questions. Nevertheless, the continually updated model generates its own pseudo-rehearsal Q&As and thus degrades as it forgets. To mitigate CF, parameter isolation methods such as Mixture of Experts (MoEs) (Shazeer et al., 2017) have shown promise. Recently, Huai et al. (2025) tackled this using Dual-Router MoE (RMoE) and Dynamic Momentum MoE (MMoE). However, their shared-parameter design is prone to degraded knowledge retention and does not naturally support task composition. At the same time, scaling up MLLMs has been shown to enhance performance (Li et al., 2023). Yet, their large parameter counts make retraining from scratch or incorporating new tasks through multiple experts computationally impractical. An alternative line of research, Visual Question Generation (VQG) (Li et al., 2018) leverages the generative capabilities of MLLMs to produce meaningful questions, while recent works show that prior knowledge can be efficiently reused across tasks (Ye & Bors, 2025; 2026; Ge et al., 2025). We exploit MLLMs' generative capacity to preserve and transfer knowledge without storing raw exemplars, using dual-purpose task experts to support reasoning, composition, and retention.

3 PRELIMINARIES

We cast visual question answering (VQA) as a generative question answering task in a continual-learning (CL) setting. Unlike conventional offline training regimes, real-world systems must handle tasks that arrive sequentially, adapting to each without retraining from scratch. This sequential formulation is particularly challenging because the model must integrate new skills while retaining prior ones, without explicit access to past data. Formally, let $\mathcal{F}_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{A}$ denote a pre-trained multimodal large language model (MLLM) with parameters $\boldsymbol{\theta}$ that maps an input space $\mathcal{X}$ of image–question pairs to a free-form textual answer space $\mathcal{A}$. The model encounters T tasks (e.g., recognition, counting) that arrive one at a time. At step t, it is given only the current task $\mathcal{T}^t$, defined by the dataset $\mathcal{D}^t = \left\{ \left((\mathbf{v}_i^t,\, q_i^t),\, a_i^t \right) \right\}_{i=1}^{|\mathcal{D}^t|}$, where $(\mathbf{v}_i^t, q_i^t)$ is an image–question pair and $a_i^t = (a_{i,1}^t, \ldots, a_{i,|a_i^t|}^t)$ is its tokenized answer sequence. The objective is to adapt $\mathcal{F}_{\boldsymbol{\theta}}$ to each new $\mathcal{T}^t$ without catastrophic forgetting of previous tasks, despite training only on $\mathcal{D}^t$ at each step. The model is finetuned sequentially on task $t = 1, \ldots, T$ with the language modeling loss $\mathcal{L}$:

$$\mathcal{L}(\boldsymbol{\theta}) \;=\; \frac{1}{|\mathcal{D}^t|} \sum_{i=1}^{|\mathcal{D}^t|} \left[-\frac{1}{|a_i^t|} \sum_{k=1}^{|a_i^t|} \log P_{\boldsymbol{\theta}}\big(a_{i,k}^t \,\big|\, a_{i,<k}^t,\, \mathbf{v}_i^t,\, q_i^t\big) \right] \tag{1}$$

where $a_{i,<k}^t$ denotes the prefix tokens of the answer up to position $k-1$. Importantly, during evaluation, the task identity (e.g., recognition) is *unknown*, making the continual setting more challenging than task-aware formulations.

4 METHODOLOGY

We present Mixture-of-Experts (MoE) framework for VQA in CL that jointly maximizes knowledge retention and enables compositional reasoning. Our method, illustrated in App. Fig. 5, unifies two objectives within an MoE framework: (i) parameter isolation for stable knowledge retention, and (ii) knowledge acquisition through a Socratic-style questioning strategy to encourage compositional reasoning. Additionally, we introduce a language-based router for task-agnostic expert selection.

4.1 KNOWLEDGE RETENTION VIA PARAMETER ISOLATION

Inspired by the success of low-rank adaptation (LoRA) (Hu et al., 2022) in large language models, we extend LoRA to MLLMs to construct task-specific experts within an MoE framework. Unlike prior approaches that combine parameter isolation with regularized parameter sharing, which often reintroduce forgetting through shared bottlenecks (Wang et al., 2024b; Christianos et al., 2021), our design enforces full isolation between experts. Each expert remains independent during continual training. This allows task-specific knowledge to be preserved while remediating catastrophic forgetting. We train each expert using the language modeling loss in Eq. (1), reformulated as

$$\mathcal{L}_{\text{VQA}}^t(\boldsymbol{\Phi}^t) = \frac{1}{|\mathcal{D}^t|} \sum_{i=1}^{|\mathcal{D}^t|} \left[-\frac{1}{|a_i^t|} \sum_{k=1}^{|a_i^t|} \log P_{\tilde{\boldsymbol{\theta}}}\big(a_{i,k}^t \,\big|\, a_{i,<k}^t,\, \mathbf{v}_i^t,\, q_i^t\big) \right], \tag{2}$$

When a new task $\mathcal{T}^{t+1}$ arrives, its parameters $\boldsymbol{\Phi}^{t+1}$ are initialized from $\boldsymbol{\Phi}^t$, but only $\boldsymbol{\Phi}^{t+1}$ is optimized while all previous experts remain frozen (warm-start only affects initialization). This prevents destructive interference and allows the expanding architecture to resist catastrophic forgetting naturally. Over time, the model accumulates low-rank specialized experts while continuing to benefit from the shared backbone.

However, because expanding architectures lack access to historical data, they receive no explicit joint supervision (Parisi et al., 2019). As a result, task-agnostic inference is challenging since task identity is unknown at test time (Hsu et al., 2018). Moreover, although prior experts remain frozen, initializing new experts from previous ones does not fully preserve prior knowledge (Schwarz et al., 2018). Without past-task data, the current expert can drift toward the new task distribution, leading to partial overwriting (Yoon et al., 2018). Approaches based on parameter sharing or past experience replay can mitigate this issue, but they introduce additional trade-offs, including interference and memory constraints. In practice, those design choices lead experts either to forget past tasks or

to over-specialize on the current task, limiting cross-task interaction and hindering the emergence of compositional behaviors, unless explicit mechanisms promote expert-to-expert guidance (Chen et al., 2021; Shazeer et al., 2017).

4.2 KNOWLEDGE COMPOSITION VIA SOCRATIC EXPERT INTERACTION

The parameter-isolated MoE architecture in Section 4.1 structurally protects the parameters of previously stored experts from subsequent updates. However, parameter isolation also limits direct knowledge sharing across experts. Because each expert is trained primarily for its corresponding task, isolation alone does not provide a mechanism for combining skills when answering compositional queries that require multiple capabilities (e.g., counting *and* color, or action *and* location).

To address the aforementioned limitation, we introduce an *inter-expert knowledge fusion* mechanism based on a principle akin to *Socratic questioning*: earlier experts "ask" the new expert structured questions that reflect their own task semantics. In this formulation, all previous experts $\{E^m\}_{m<t}$ serve as frozen teachers, while the new expert E^t is the student. The objective is not only to preserve earlier abilities, but also to expose E^t to how previous experts interrogate and interpret the current images, enabling it to generalize compositionally across tasks that were never observed jointly.

To build a unified framework, we adopt the same architecture for the Socratic questioning (i.e., visual question generation (VQG)) module as for the VQA module, ensuring a consistent, streamlined design without introducing unnecessary complexity. We extend our expert definition to incorporate both components, denoting each expert as $E^t = \{\{\mathcal{E}^t_{VQA}, \mathcal{E}^t_{VQG}\}, \quad \boldsymbol{\psi}^t = \{\boldsymbol{\Phi}^t_{VQA}, \boldsymbol{\Phi}^t_{VQG}\}\}$, where $\mathcal{E}^t_{VQA}$ and $\mathcal{E}^t_{VQG}$ are the task-$\mathcal{T}^t$ dual purpose components, respectively, and $\boldsymbol{\Phi}^t_{VQA}$, $\boldsymbol{\Phi}^t_{VQG}$ are their corresponding low-rank parameters. The VQG component is trained jointly on the same task dataset as the VQA component, but with the input–output roles reversed. Specifically, instead of receiving a question and predicting an answer, the expert takes a text prompt and an image as input and generates the corresponding question. During VQG training, we use a structured prompt of the form p^t = `Generate a` $\{\mathcal{T}^t\}$ `question for the image, where the possible answers are` $\{a^t\}$. The ground-truth questions from the dataset serve as the output target, and the expert is optimized with the language modeling loss defined in Eq. (3).

$$\mathcal{L}^t_{VQG} \;=\; \frac{1}{|\mathcal{D}^t|}\sum_{i=1}^{|\mathcal{D}^t|}\left[-\frac{1}{|q^t_i|}\sum_{k=1}^{|q^t_i|}\log P_{\tilde{\boldsymbol{\theta}}}\big(q^t_{i,k} \,\big|\, q^t_{i,<k},\, \mathbf{v}^t_i,\, p^t\big)\right], \tag{3}$$

Here, $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}\big(\boldsymbol{\Phi}^t_{VQG}\big)$, and q^t_i=$(q^t_{i,1}, \ldots, q^t_{i,|q^t_i|})$ denotes the tokenized question sequence. Together, Eqs. (2) and (3) ensure that task-specific knowledge of E^t is preserved. We define the per-task objective as the weighted sum of the VQG and VQA losses in Eq. (4). While this constitutes a joint learning objective, the two expert branches are optimized in an alternating manner in practice, where $\lambda_{\text{VQG}}, \lambda_{\text{VQA}} \geq 0$ are balancing coefficients.

$$\min_{\Phi^t_{\text{VQG}},\, \Phi^t_{\text{VQA}}} \mathcal{L}^t_{\text{joint}} = \lambda_{\text{VQG}}\mathcal{L}^t_{\text{VQG}} + \lambda_{\text{VQA}}\mathcal{L}^t_{\text{VQA}}. \tag{4}$$

At step t, each previous expert E^m $(m < t)$ generates QA pairs $(\tilde{q}^m, \tilde{a}^m)$ on current-task images. During Socratic generation, we use the task-conditioned prompt p^m_{gen} in App. E. To ensure the quality of the generated questions, we employ confidence-based filtering. For each generated question $\tilde{q}^m_i = (\tilde{q}^m_{i,1}, \ldots, \tilde{q}^m_{i,|\tilde{q}^m_i|})$, we define its confidence as the length-normalized log-likelihood score:

$$C(\tilde{q}^m_i) = \frac{1}{|\tilde{q}^m_i|}\sum_{k=1}^{|\tilde{q}^m_i|}\log P_{\boldsymbol{\theta},\boldsymbol{\Phi}^m_{VQG}}\big(\tilde{q}^m_{i,k} \,\big|\, \tilde{q}^m_{i,<k},\, \mathbf{v}^t_i,\, p^m_{\text{gen}}\big)\,. \tag{5}$$

We then rank generated questions by $C(\tilde{q}^m_i)$ in descending order and retain only the top $\rho\%$:

$$\mathcal{Q}^m_{\text{filtered}} = \text{Top}_{\rho\%}(\{\tilde{q}^m_i\},\, C)\,. \tag{6}$$

This filtering step removes low-confidence generations. We further ensure that the questions are answerable and apply additional quality controls (see App E). Finally, for each question in $\mathcal{Q}^m_{\text{filtered}}$, we associate it with its corresponding image and leverage the corresponding $\mathcal{E}^m_{VQA}$ to generate answers. Each generated pair is appended as an additional conversation turn to the training example for that image, so an image already present in the current task $\mathcal{D}^t$ simply gains extra question-answer turns from earlier experts without repeating the image, while the current expert is optimized using Eq. (2). Specifically, VQG generates semantic questions that serve as probes of current-task

images, revealing how prior experts decompose and interpret the same visual scene. By training on these expert-derived perspectives, the student expert learns to align and integrate previously isolated skills, enabling richer cross-task interactions. In this sense, generation is used to promote knowledge composition rather than rehearsal. App Fig. 4 illustrates this Socratic multi-view supervision: while standard VQA provides a single question–answer pair per image, our framework augments each image with diverse expert-generated views. Learning from these complementary perspectives encourages the model to integrate multiple task-specific factors (see Tab. 4).

4.3 Task-agnostic Inference

While frozen task-specific experts preserve previously learned knowledge, they introduce inference-time challenge in CL: the model must select an appropriate expert without access to the task identity. We address this with a lightweight language-based router. The router is used only for task-agnostic expert selection; compositional knowledge is acquired during training through Socratic expert interaction (Sec 4.2), rather than by combining multiple expert outputs at inference time.

The motivation for language-only routing is that the question form often provides strong cues about the dominant answering skill. For example, questions asking "how many" typically require counting expertise, whereas questions asking "what color" require color recognition. In contrast, image features are less discriminative for routing because the same image may support many task types. For instance, "What color is the man's shirt?" and "What is the man doing?" may refer to the same visual region but require different experts. We therefore route based on the question text alone. Given a question sequence q, the router encodes it into a latent representation h_q and maps it to a distribution over the experts available after step t: $r(q) = \mathrm{softmax}\left(W_{\mathrm{fc}}^{t} h_q\right)$, where W_{fc}^{t} is the router classification head at continual-learning step t. The selected expert is $j^{\star}(q) = \arg\max_{j \in 1,\ldots,t} r(q)[j]$. As new tasks arrive, the router is incrementally expanded by adding one output logit for each new expert. At step t, the router is trained on $\tilde{\mathcal{Q}}^{t} = \{(q,t) : q \in \mathcal{Q}^{t}\} \cup \bigcup_{m<t} \{(q,m) : q \in \mathcal{Q}^{m}_{\mathrm{filtered}}\}$, i.e., the current task questions labeled t together with the filtered questions generated by each previous expert $\mathcal{E}^{m}_{VQG}$ (Eq. (6)), labeled m, using the standard cross-entropy objective:

$$\mathcal{L}_{\mathrm{router}} = -\frac{1}{|\tilde{\mathcal{Q}}^{t}|} \sum_{(q,\,y) \in \tilde{\mathcal{Q}}^{t}} \log r(q)[y], \tag{7}$$

where y is the task label of q. This formulation follows the task-agnostic continual VQA setting. We provide the overall architecture of our method in Appendix Fig. 5.

5 Compositional VQA Benchmark

The VQACL (Zhang et al., 2023) setting assumes a predefined task boundary determined by linguistically defined tasks. However, in real-world applications, evaluating open-ended models under strictly isolated task settings is unrealistic. For instance, consider a robotics assistant in a home environment: when asked to "count the number of blue cups on the table," the system must simultaneously interpret and compose the count and color tasks. Current isolated-task evaluations fail to capture such task composition. Existing benchmarks define composition from different perspectives, including object composition (Zhang et al., 2023), compositions of seen concepts (Agrawal et al., 2017), and scene-level composition (Hudson & Manning, 2019). Unlike prior work, none of these benchmarks deliberately compose multiple linguistically distinct tasks within the CL paradigm. We introduce **human-generated linguistic task composition** within a CL setting, addressing this gap.

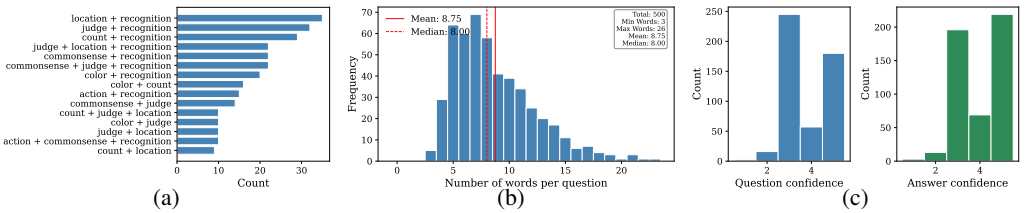


Figure 2: Overall statistics of our Compositional VQA dataset. (a) Distribution of composition types (top 15), (b) Distribution of word counts in questions, (c) Distribution of self-reported question-answer confidence (1:low, 5:high).

5.1 PIPELINE

Our dataset construction pipeline unfolds through several stages, each designed to address key practical and methodological challenges in CL and to ensure annotation quality. First, we collect images to anchor our benchmark in realistic visual diversity, using all 5,000 test-set images from the VQACL benchmark together with their corresponding task, question, and answer triples. This selection provides a rich foundation and enables direct comparison with prior continual VQA works.

To facilitate efficient QA collection, we developed a custom web-based annotation platform tailored to our requirements (App Fig. 7). The interface contains three main components:

1. **Image view:** presents a single image to annotators for analysis and composition.
2. **Existing triples panel:** lists all corresponding VQACL task–question–answer triples for the image, equipping annotators with context and highlighting previous interpretations.
3. **Annotation form:** enables annotators to craft new compositional question–answer pairs by actively choosing two or more tasks that the composed question integrates.

Annotators were free to formulate open-ended compositional questions; existing QA triples served only as guidance rather than constraints. Images could be skipped, as not all support the construction of compositional questions. In addition, we collect annotators' self-reported confidence scores that reflect both the quality of the question and the reliability of the answer. These confidence scores are not merely auxiliary signals; they are directly used in subsequent evaluation and analysis. For example, we use the confidence ratings to filter out low-confidence examples, ensuring that the evaluation focuses on high-quality data. Figure 2(c) shows question-answer self-reported confidence distributions.

We recruited sixteen students from diverse educational and disciplinary backgrounds, each assigned a dedicated annotation account. Before annotation, all participants completed a training session covering task definitions, composition guidelines, and expected quality standards. To further improve annotation reliability and transparency, after the main annotation phase, annotators reviewed a random subset of peer submissions and rated the quality of both questions and answers. We filter out 46 questions that received low confidence scores from both annotators and reviewers, as well as 8 annotations containing either incomplete sentences or duplicate entries. Our final dataset comprises 500 annotations across 418 unique images, spanning 100 distinct composition types. Figure 2 shows the overall statistics of our dataset. See App Fig. 6 for additional statistics.

Table 1: Performance comparison across ten tasks. Average performance (AP) after all tasks. Best in **bold**, second best underlined.

Methods	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Cau.	AP (↑)
Vanilla	19.25	14.81	54.59	56.97	24.23	46.20	27.58	26.09	36.47	18.89	32.51
EWC (Kirkpatrick et al., 2017)	28.12	23.02	61.50	61.08	26.13	54.29	23.65	32.25	44.97	17.83	37.28
MAS (Aljundi et al., 2018)	31.54	22.09	60.85	46.32	32.48	56.47	30.05	35.69	42.73	18.83	37.71
ER (Chaudhry et al., 2019)	29.31	25.74	63.46	65.78	31.92	58.39	45.17	34.55	46.24	18.96	41.95
DER (Buzzega et al., 2020)	26.95	21.43	64.88	66.17	31.01	55.92	44.60	32.85	47.09	20.74	41.16
VS (Wan et al., 2022)	28.48	24.09	61.37	67.20	29.56	54.64	33.98	32.91	45.82	19.89	39.79
VQACL (Zhang et al., 2023)	34.14	32.19	66.15	63.00	33.01	60.91	34.64	38.48	47.94	24.42	43.49
QUAD (Marouf et al., 2025)	43.50	32.19	70.97	72.05	41.10	68.85	61.73	55.08	58.17	**30.41**	53.40
GaB (Das et al., 2025)	47.20	33.33	76.34	71.15	43.41	70.65	63.12	56.48	57.04	26.27	54.50
CL–MoE (Huai et al., 2025)	46.50	37.18	75.22	71.39	40.90	69.54	43.66	52.68	55.55	20.74	51.34
Ours	**56.66**	**44.02**	**81.37**	**74.49**	**53.15**	**75.55**	**74.79**	**61.85**	**59.66**	**30.41**	**61.20**
Joint Training	57.29	45.01	83.19	78.00	52.98	78.31	74.95	63.34	61.85	29.03	-
Oracle	68.30	55.13	97.67	96.82	70.64	90.61	83.35	76.05	74.88	44.70	-

6 EXPERIMENTS

Backbone & Baselines & Metrics. We evaluate using two representative MLLM backbones: LLaVA-7B (Liu et al., 2023) and BLIP-2 (Li et al., 2023). We benchmark performance under standard continual learning baselines. For replay-based methods, we consider ER (Chaudhry et al., 2019), DER (Buzzega et al., 2020), and VS (Wan et al., 2022), while for regularization-based methods, we include EWC (Kirkpatrick et al., 2017) and MAS (Aljundi et al., 2018). In addition, we compare against frameworks designed specifically for open-ended language generation, including

Table 2: CLIP cosine similarity statistics. See App. Tab. 12.

Metric	Avg. cosine (↑)
Real Ques.	0.24±0.03
Generated Ques.	**0.25±0.03**
Shuffled Neg. Ques.	0.17±0.02
δ (10th pct. Real)	0.19
%> δ Real Ques.	90.0
%> δ Generated Ques.	**97.0**
%> δ Shuffled Neg. Ques.	21.2

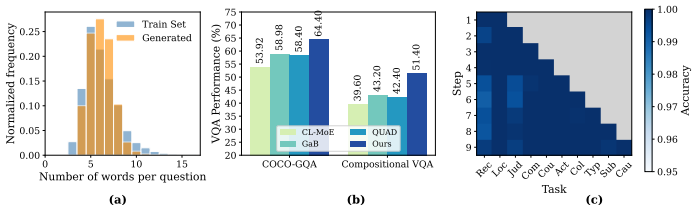


Figure 3: (a) Lexical statistics (b) Compositional performance (c) Router robustness

VQACL (Zhang et al., 2023), GaB (Clustering) (Das et al., 2025), QUAD (Marouf et al., 2025), and CL-MoE (Huai et al., 2025). We include Vanilla, which refers to incremental training without any continual learning method. Joint Training means training all tasks together. We report a post hoc output oracle that selects the expert whose answer matches the ground truth if one exists. It relies on answer correctness after generation; thus, it is not a routing upper bound. We adopt the standard continual learning metric, Final Average Performance (AP) (Zhang et al., 2023), defined as $AP = \frac{1}{T}\sum_{t=1}^{T} \alpha_{T,t}$, where $\alpha_{i,j}$ denotes the test performance on task j after completing training on task i, and T denotes the total number of tasks. We use publicly available results when available.

6.1 Main Results on the Traditional VQACL

Setting. Following prior continual VQA literature, we conduct our experiments in the VQACL setting (Zhang et al., 2023), which is derived from the VQA v2 benchmark (Goyal et al., 2017). The benchmark is partitioned into ten linguistically driven tasks: *Recognition*, *Location*, *Judge*, *Commonsense*, *Count*, *Action*, *Color*, *Type*, *Subcategory*, and *Causal*.

Analysis. We report the main results in Tab. 1. Additionally, results on BLIP-2 (5B) are provided in the App. Tab 8. The VQACL benchmark shows a clear difficulty spectrum across tasks, with causal reasoning consistently the most challenging and yielding the lowest accuracy. Our method achieves the highest AP (61.20% for LLaVA-7B) while remaining competitive across individual tasks, demonstrating effective knowledge preservation via parameter isolation. We also note that the conventional forgetting metric is less informative in our MoE design, since routing to OOD experts may still produce valid answers, leading to negative forgetting values that reflect metric limitations rather than true forgetting. Additionally, because trained experts are frozen, later tasks cannot overwrite their task-specific parameters; remaining errors mainly arise from routing mistakes or limited expert competence, rather than parameter drift in previous experts.

6.2 Results on the Compositional VQA for CL

Setting. We assess compositional performance using two complementary benchmarks. First, we construct **COCO-GQA** from the subset of GQA (Hudson & Manning, 2019) whose images overlap with the VQACL test split. While GQA questions are generated programmatically from structured scene graphs via an automated question engine, the dataset provides large-scale coverage of multi-step visual reasoning. To mitigate the synthetic bias of GQA, we additionally evaluate on our human-annotated **compositional VQA** dataset (Sec. 5), which emphasizes natural-language compositionality beyond existing benchmarks.

Analysis. In Fig. 3(b), we report compositional evaluation results using the LLaVA-7B backbone. Our method achieves the best performance in both settings. On COCO-GQA, it reaches 64.40%, outperforming the next-best method, GaB, at 58.98% by 5.42 percentage points. On the human-annotated compositional VQA benchmark, our approach again leads with 51.40%, compared with 43.20% for GaB and 42.40% for QUAD. As shown in Fig. 1, the MoE-based CL-MoE is limited by its focus on per-task expertise, while GaB and QUAD primarily emphasize knowledge retention within a shared learning space, which can remain susceptible to forgetting. In contrast, our approach combines multi-view supervision with parameter isolation, maintaining strong performance across both synthetic and human-annotated compositional settings.

6.3 Ablation Study

Performance of Router. We evaluate the router under task-agnostic inference, as required in the CL setting where task IDs are unavailable at inference, rather than assuming known task IDs as in

multi-task learning, and treat task identity prediction as the target. Figure 3(c) reports routing accuracy across CL steps. The router consistently maintains near-perfect accuracy throughout training, demonstrating stable task discrimination and robustness without access to task IDs at test time.

Analysis of Generated Questions. We compare lexical statistics between generated and original questions. As shown in Fig. 3(a), the length distribution of generated questions is similar to that of the training set. Additionally, we compute CLIP cosine similarity (Zhengwentai, 2023) between images and questions (Tab. 2). Using the 10th percentile of real-pair scores as the threshold (δ), 95–98% of generated questions exceed it, compared with only 16–33% of shuffled pairs, suggesting strong image–question alignment. We also conduct a blind human evaluation of three question groups using a masked 1–5 rating scale, with group labels hidden from raters and 51 randomly sampled examples from each group, as shown in Tab. 3. The results suggest that the generated questions approach human-rated quality.

Table 3: Human evaluation ratings.

	Real	**Retained Gen.**	**Filtered-out Gen.**
Rating↑	4.04	3.67	3.22

Replay vs. Socratic Questioning. We design a controlled setting by keeping the number of samples identical, using a single expert with an LLaVA-7B backbone on the VQACL (Zhang et al., 2023) benchmark. We compare (i) complementary views of different tasks (i.e., Q&As) on the current task images, and (ii) samples from different tasks paired with their corresponding images. Thus, both settings use the same number of Q&As, but the latter setting uses additional images from past tasks. Keeping all other settings fixed, we treat the last task (causal) as the current task and the remaining tasks as previous ones. Table 4 shows that using the same image with complementary task views improves overall performance on both traditional and the compositional setting.

Table 4: Replay vs. Socratic questioning.

Method	VQACL	COCO-GQA	Image
Replay	49.61	60.80	Current + Past
Socratic Q.	**51.45**	**61.76**	Current only

Effect of Socratic questioning. To assess the impact of the Socratic component, we remove it from our framework while keeping the same expert pool and router. Table 5 shows that removing it reduces AP from 61.20% to 58.54% and COCO-GQA accuracy from 64.40% to 54.53%. The larger drop on compositional questions (9.87 percentage points) indicates that Socratic multi-view supervision primarily improves cross-task composition rather than per-task accuracy.

Table 5: Effect of Socratic questioning.

Method	VQACL	COCO-GQA
w/o Socratic	58.54	54.53
Ours	**61.20**	**64.40**

Retention Ratio. Table 6 shows the effect of retention ratio ρ (Eq. (6)) on VQACL average performance (AP). A smaller ρ improves efficiency but limits knowledge fusion, while a larger ρ increases coverage at the cost of added noise and computation. $\rho = 50\%$ yields the best trade-off.

Table 6: Effect of ρ.

ρ	25%	50%	75%
AP↑	60.95	61.20	60.48

Storage Footprint vs. Replay. Table 7 compares the storage required by a replay buffer of 5,000 image–question pairs (∼770 MB) with that required by our task-specific experts (∼233 MB per task). Prior work shows that the performance of replay-based CL scales with replay size (Balaji et al., 2020), which increases the amount of past data that must be retained. This distinction matters for privacy, as VQA data may expose sensitive information through both images (Gurari et al., 2019) and text (Tito et al., 2024). In contrast, although our pool of frozen experts grows as new tasks arrive, each consists of low-rank parameters rather than stored raw exemplars. Thus, our method avoids explicit retention of past raw data. This footprint could be further reduced through adapter compression (e.g., quantization; see Appendix I).

Table 7: Storage footprint.

Method	Storage	Stored Content
Replay buffer	∼770 MB	Raw exemplars
Ours	∼233 MB	Model parameters

7 Conclusion

In this paper, we introduce dual-purpose experts within a Mixture-of-Experts (MoE) framework for compositional continual VQA. Parameter isolation retains past knowledge, while Socratic questioning between experts transfers knowledge across tasks and improves compositional generalization, without storing raw exemplars. A lightweight language-based router enables task-agnostic inference. We also introduce a human-annotated benchmark for task compositionality in continual VQA. Experiments on VQACL, COCO-GQA, and our compositional benchmark show that our approach outperforms prior continual VQA methods. Our method currently composes only previously learned tasks; extending it to compositions involving future tasks is a promising direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing code and polishing the writing. We did not use generative AI tools for the other tasks requiring disclosure, and the remaining required disclosure tasks were not applicable to this work. All AI-assisted code was reviewed, tested, and verified by the authors, and all AI-assisted writing was reviewed and revised by the authors. The research methodology and proposed method were developed by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our compositional VQA benchmark was annotated by sixteen student volunteers who received task training, could skip any image, and were free to leave the project at any time. The annotations use publicly available Microsoft COCO: Common Objects in Context images; no personal information about the annotators was collected or released. The work does not meet the regulatory definition of "human subjects research," and review and approval by the anonymous institution's IRB were not required.

REFERENCES

Aishwarya Agrawal, Aniruddha Kembhavi, Dhruv Batra, and Devi Parikh. C-vqa: A compositional split of the visual question answering (vqa) v1.0 dataset, 2017. URL https://arxiv.org/abs/1704.08243.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 139–154, 2018.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, et al. Qwen technical report, 2023. URL https://arxiv.org/abs/2309.16609.

Yogesh Balaji, Mehrdad Farajtabar, Dong Yin, Alex Mott, and Ang Li. The effectiveness of memory replay in large scale continual learning. *arXiv preprint arXiv:2010.02418*, 2020.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, P Dokania, P Torr, and M Ranzato. Continual learning with tiny episodic memories. In *Workshop on Multi-Task and Lifelong Reinforcement Learning*, 2019.

Wenhu Chen, Zhe Gan, Linjie Li, Yu Cheng, William Wang, and Jingjing Liu. Meta module network for compositional visual reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 655–664, 2021.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Filippos Christianos, Georgios Papoudakis, Muhammad A Rahman, and Stefano V Albrecht. Scaling multi-agent reinforcement learning with selective parameter sharing. In *International Conference on Machine Learning*, pp. 1989–1998. PMLR, 2021.

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

Deepayan Das, Davide Talon, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. One vlm to keep it learning: Generation and balancing for data-free continual visual question answering. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 5635–5645. IEEE, 2025.

Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19358–19369, 2023.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Chendi Ge, Xin Wang, Zeyang Zhang, Hong Chen, Jiapei Fan, Longtao Huang, Hui Xue, and Wenwu Zhu. Dynamic mixture of curriculum loRA experts for continual multimodal instruction tuning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=zpGK1bOlHt.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Haiyang Guo, Fanhu Zeng, Fei Zhu, Jiayi Wang, Xukai Wang, Jingang Zhou, Hongbo Zhao, Wenzhuo Liu, Shijie Ma, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin Liu. Continual learning for generative ai: From llms to mllms and beyond, 2025. URL https://arxiv.org/abs/2506.13045.

Danna Gurari, Qing Li, Chi Lin, Yinan Zhao, Anhong Guo, Abigale Stangl, and Jeffrey P. Bigham. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 939–948, 2019. doi: 10.1109/CVPR.2019.00103.

Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. In *NeurIPS 2018 Workshop on Continual Learning*, 2018. URL https://arxiv.org/abs/1810.12488.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Tianyu Huai, Jie Zhou, Xingjiao Wu, Qin Chen, Qingchun Bai, Ze Zhou, and Liang He. Cl-moe: Enhancing multimodal large language model with dual momentum mixture-of-experts for continual visual question answering. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 19608–19617, 2025.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6700–6709, 2019.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128–3137, 2015.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

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

Yikang Li, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. Visual question generation as dual task of visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6116–6124, 2018.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

Imad Eddine Marouf, Enzo Tartaglione, Stéphane Lathuilière, and Joost Van De Weijer. Ask and remember: A questions-only replay strategy for continual visual question answering. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18078–18089. IEEE, 2025.

OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

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

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks, 2022. URL https://arxiv.org/abs/1606.04671.

Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International conference on machine learning*, pp. 4528–4537. PMLR, 2018.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras. In *European Conference on Computer Vision*, pp. 422–438. Springer, 2024.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, et al. Openai gpt-5 system card, 2026. URL https://arxiv.org/abs/2601.03267.

Vishal Thengane, Salman Khan, Munawar Hayat, and Fahad Khan. Clip model is an efficient continual learner, 2022. URL https://arxiv.org/abs/2210.03114.

Rubèn Tito, Khanh Nguyen, Marlon Tobaben, Raouf Kerkouche, Mohamed Ali Souibgui, Kangsoo Jung, Joonas Jälkö, Vincent Poulain D'Andecy, Aurelie Joseph, Lei Kang, Ernest Valveny, Antti Honkela, Mario Fritz, and Dimosthenis Karatzas. Privacy-aware document visual question answering. In Elisa H. Barney Smith, Marcus Liwicki, and Liangrui Peng (eds.), *Document Analysis and Recognition - ICDAR 2024*, pp. 199–218, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-70552-6.

Timmy ST Wan, Jun-Cheng Chen, Tzer-Yi Wu, and Chu-Song Chen. Continual learning for visual search with backward consistent feature embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16702–16711, 2022.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024a. doi: 10.1109/TPAMI.2024.3367329.

Mingyang Wang, Heike Adel, Lukas Lange, Jannik Strötgen, and Hinrich Schuetze. Rehearsal-free modular and compositional continual learning for language models. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 469–480, Mexico City, Mexico, June 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-short.39. URL https://aclanthology.org/2024.naacl-short.39/.

Spencer Whitehead, Hui Wu, Heng Ji, Rogerio Feris, and Kate Saenko. Separating skills and concepts for novel visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5632–5641, 2021.

Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Philip S Yu. Multimodal large language models: A survey. In *2023 IEEE International Conference on Big Data (BigData)*, pp. 2247–2256. IEEE, 2023.

Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, Xiaopeng Zhang, and Qi Tian. Qa-lora: Quantization-aware low-rank adaptation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 52401–52418, 2024.

Fei Ye and Adrian G Bors. Online continual learning via dynamic expandable recursive model. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 8087–8096, 2025.

Fei Ye and Adrian G Bors. Online task-free continual learning via expansible vision transformer. *Pattern Recognition*, 169:111730, 2026.

Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Sk7KsfW0-.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, et al. Opt: Open pre-trained transformer language models, 2022. URL https://arxiv.org/abs/2205.01068.

Xi Zhang, Feifei Zhang, and Changsheng Xu. Vqacl: A novel visual question answering continual learning setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19102–19112, 2023.

SUN Zhengwentai. clip-score: CLIP Score for PyTorch. https://github.com/taited/clip-score, March 2023. Version 0.2.1.

A Appendix

We organize the appendix to provide supplementary material to support the main paper. We first present additional experimental results and ablation studies, including extended BLIP-2 results, expert-pool analysis, and expert merging. We then provide further details on the expert architecture for both the BLIP-2 and LLaVA backbones, followed by a description of the question-generation pipeline and CLIP-based image–question alignment. Next, we describe the benchmark construction process for VQACL, COCO-GQA, and our human-annotated Compositional VQA benchmark. Finally, we report implementation details, workstation setup, limitations, and reproducibility information.

B Additional Results

As described in Sec. 6, additional results on BLIP-2 (5B) are provided by extending Tab. 1 from the main paper to Tab. 8 in this Appendix. In Tab 8, we report a post-hoc output oracle as a non-deployable diagnostic upper bound. For each sample, this oracle selects the expert whose generated answer matches the ground-truth answer, if such an expert exists. This setting is not a routing upper bound because it relies on answer correctness after generation rather than on task identity or router scores; we include it only to estimate the remaining headroom in the expert pool. We include replay based and also generative replay based baseline such as GaB. Overall, Tab. 8 shows that our method achieves the highest AP among the evaluated BLIP-2 baselines (ER, DER, GaB, and VQACL), consistent with the LLaVA-7B results in Tab. 1.

Table 8: Performance comparison across ten tasks (BLIP-2 backbone). Average performance (AP) after all tasks. Best in **bold**, second best underlined.

Methods	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Cau.	AP (↑)
BLIP-2 backbone											
ER (Chaudhry et al., 2019)	45.76	25.64	72.87	66.91	15.35	67.12	53.28	50.70	49.87	15.20	46.27
DER (Buzzega et al., 2020)	44.87	24.06	71.12	65.41	38.78	60.52	46.17	36.77	48.90	14.90	45.15
GaB (Das et al., 2025)	28.03	29.05	**79.99**	**77.80**	45.29	65.30	59.96	32.30	50.02	11.60	47.93
VQACL (Zhang et al., 2023)	**47.92**	25.70	75.95	69.86	41.42	64.63	49.31	39.27	52.22	15.92	48.22
Ours	47.78	**35.02**	79.43	74.18	**48.23**	**73.20**	**69.53**	**56.38**	**57.06**	**20.00**	**56.08**
Oracle	58.42	47.72	90.28	88.02	61.58	83.36	77.48	64.08	64.09	31.34	-

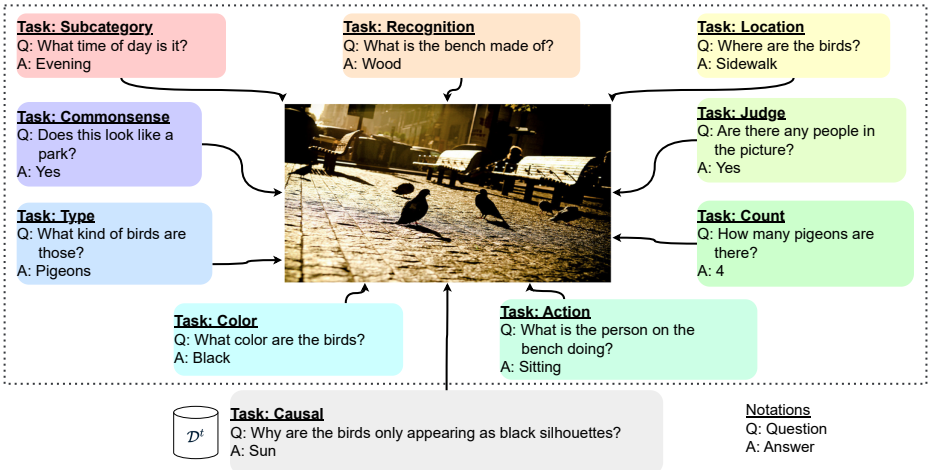


Figure 4: **Socratic questioning**. Traditional VQA relies on a single-step learning setting (gray outer box), where each training instance is a triple $\left((\mathbf{v}_i^t, , q_i^t), , a_i^t\right) \in \mathcal{D}^t$. Our method extends this into a compositional space (dotted box) by querying multiple diverse Q&A types for the same image. These question types provide complementary semantic views, enabling Socratic, multi-view supervision, fostering richer multimodal reasoning and stronger **compositional skills**. Distinct colors denote different tasks.

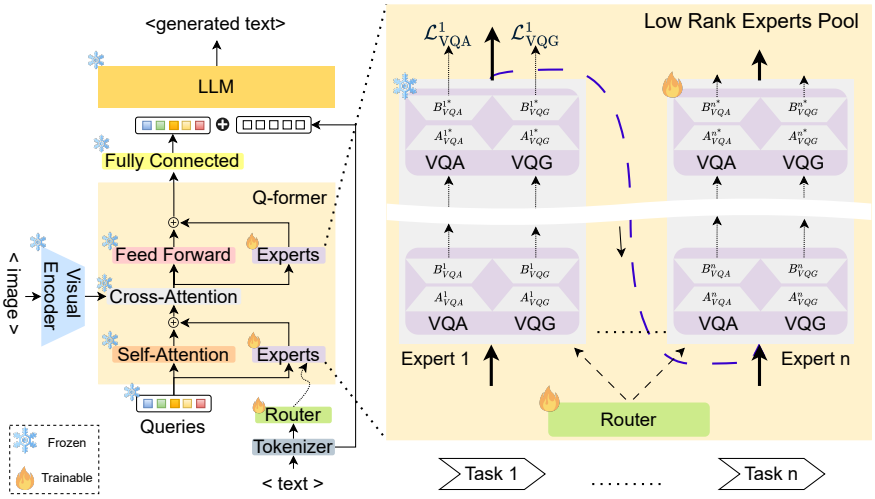


Figure 5: Overview of the proposed dual-purpose MoE framework. For both VQA and VQG, low-rank adapters are applied in parallel to the self-attention and feedforward layers. In the cross-attention layer, the experts' outputs, combined with the frozen self-attention output, serve as the cross-attention query, thereby helping extract task-specific features from the input image. Collectively, we refer to this as an Expert. During training, we optimize the current expert and the LSTM-based router while all other components remain frozen. Here we demonstrate the QFormer of BLIP-2 as a multimodal space.

C Additional Ablations

We ablate different design choices: whether merging multiple experts is beneficial, whether routing over the full expert pool is necessary, and whether Socratic questioning helps in composition.

Merging multiple experts at inference degrades performance. Our expert pool retains learned knowledge and promotes composition during training. However, a recent study, ZipLoRA (Shah et al., 2024), shows that multiple LoRAs can be merged; however, this depends on task orthogonality. We choose ZipLoRA-style merging for this ablation, where two LoRAs with different learned capabilities are merged. Due to the natural restriction of merging at most two LoRAs in ZipLoRA, we choose top-2 routed expert merging using our router from Sec. 4.3. Additionally, ZipLoRA requires merger coefficient training, which in our case requires samples from both tasks (i.e., replay samples), so to ensure a fair comparison, we choose replay-free ZipLoRA-direct. Table 9 shows that task-agnostic single-expert routing beats ZipLoRA-direct merging on all 10 tasks and in AP (+4.51). We also conduct ZipLoRA-direct merging on our compositional VQA benchmark, where ZipLoRA-direct merging achieves 46.40%, while our method achieves 51.40% (+5.00).

Table 9: ZipLoRA vs. our Socratic expanding expert design. Merge results use our router and a ZipLoRA-style merge of the top-2 predicted experts (LLaVA-7B backbone). Best in **bold**.

Method	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Cau.	AP
ZipLoRA-direct	52.11	38.46	77.88	71.07	47.06	72.58	71.48	57.97	54.36	23.96	56.69
Ours	**56.66**	**44.02**	**81.37**	**74.49**	**53.15**	**75.55**	**74.79**	**61.85**	**59.66**	**30.41**	**61.20**

Final expert vs. full expert pool. We ablate if the full expert pool is necessary on Tab. 10. The final expert is trained with Socratic QA generated by all previous frozen experts, so its quality depends on the stored pool: used alone, it reaches 56.79% AP on VQACL. This final expert cannot be obtained without the pool, and the pool remains necessary to generate Socratic supervision for any future task. At inference, routing over the full pool further improves 9 out of 10 tasks (61.20% AP), as frozen prior experts retain task-specialized precision (e.g., count: 53.15 vs. 43.27). On the compositional tasks—COCO-GQA and our human-annotated compositional benchmark, denoted as Comp.—the

trend continues. Thus, the pool serves two roles: it provides cross-task supervision during training and task-specific precision at inference.

Table 10: Final expert only vs. routing over the full expert pool (LLaVA-7B). Best in **bold**.

Method	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Cau.	AP	COCO-GQA	Comp.
Final expert only	49.82	38.46	77.88	73.68	43.27	73.90	65.40	56.56	58.54	30.41	56.79	62.54	47.60
Ours (full pool)	**56.66**	**44.02**	**81.37**	**74.49**	**53.15**	**75.55**	**74.79**	**61.85**	**59.66**	30.41	**61.20**	**64.40**	**51.40**

More on Effect of Socratic questioning. We extend Tab. 5 from the main paper to per task breakdown in Tab. 11. Socratic training improves 8 out of 10 tasks. Additionally, the largest gains are observed on the compositional benchmark COCO-GQA, suggesting that Socratic questioning helps the model learn task composition.

Table 11: Per-task effect of Socratic questioning (LLaVA-7B). The baseline uses independently trained experts with our router, trained on plain task data without Socratic QA. Best in **bold**.

Method	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Cau.	AP	COCO-GQA	Comp.
w/o Socratic	**57.43**	40.18	79.91	68.29	51.40	71.20	**78.95**	57.97	53.30	26.73	58.54	54.53	45.40
Ours	56.66	**44.02**	**81.37**	**74.49**	**53.15**	**75.55**	74.79	**61.85**	**59.66**	**30.41**	**61.20**	**64.40**	**51.40**
Δ	-0.77	$+3.84$	$+1.46$	$+6.20$	$+1.75$	$+4.35$	-4.16	$+3.88$	$+6.36$	$+3.68$	$+2.66$	$+9.87$	$+6.00$

D Additional details of Expert Architecture

We define each expert as $E^t = \{\mathcal{E}^t_{VQA}, \mathcal{E}^t_{VQG}\}, \quad \boldsymbol{\psi}^t = \{\boldsymbol{\Phi}^t_{VQA}, \boldsymbol{\Phi}^t_{VQG}\}$, where $\mathcal{E}^t_{VQA}$ and $\mathcal{E}^t_{VQG}$ are the task-specific dual-purpose components, respectively, and $\boldsymbol{\Phi}^t_{VQA}$, $\boldsymbol{\Phi}^t_{VQG}$ are the purpose specific corresponding low-rank parameters. However, each component shares the same underlying architecture. Specifically, we adopt LoRA (Hu et al., 2022) as the low-rank adapter.

D.1 BLIP-2 Backbone

As described in Section 4.1, we apply an adapter in the multi-modal space of MLLMs. Now denote a given layer with input $x \in \mathbb{R}^{d_{\text{in}}}$ and weight matrix $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, the original transformation is $y = Wx$. With LoRA, an adapter introduces a low-rank update $\Delta y = A(Bx)$, where $A \in \mathbb{R}^{d_{\text{out}} \times r}$ and $B \in \mathbb{R}^{r \times d_{\text{in}}}$ are low-rank matrices ($r \ll \min(d_{\text{in}}, d_{\text{out}})$). The final output is computed as $y' = Wx + \alpha \cdot A(Bx)$, where α is a scaling factor. In our expert setting, each task $\mathcal{T}^t$ is assigned an isolated low-rank parameter space while the pre-trained backbone $\boldsymbol{\theta}$ remains frozen (i.e., both vision encoder and LLM). A component (i.e., VQA or VQG) output for task $\mathcal{T}^t$ is thus parameterized as:

$$\mathcal{E}^t(\mathbf{v}^t,\, q^t) = \mathcal{F}_{\boldsymbol{\theta}+\Delta\boldsymbol{\theta}(\boldsymbol{\Phi}^t)}(\mathbf{v}^t,\, q^t),$$

where $\Delta\boldsymbol{\theta}(\boldsymbol{\Phi}^t)$ denotes the collection of all low-rank updates applied across the designated projection and feed-forward layers.

D.2 LLaVA Backbone

We apply the same expert design to the LLaVA backbone. Specifically, we attach low-rank adapters to the linear layers of the language model. Each task $\mathcal{T}^t$ is assigned its own VQA and VQG adapter pair $\{\Phi^t_{\text{VQA}}, \Phi^t_{\text{VQG}}\}$ with projector, which is frozen after training on that task, while the pretrained backbone remains frozen throughout.

E Details of Question Generation

The questions for prior tasks are produced through a structured generate-filter-answer pipeline designed to maximize both task relevance and linguistic quality. Concretely, for each previously learned task (m), the framework performs the corresponding task VQG using a task-conditioned

prompt that (i) explicitly asks for a question focused on the task and (ii) explicitly discourages drift toward other task semantics. We use the prompt, p^m_{gen}: "Given the image, generate clear, and natural question that a human might ask focusing solely on the {TASK m} task. Ensure the question is specific, avoids ambiguity, and is distinct from other tasks such as {OTHER_TASKS}." Given an input image ($\mathbf{v}_i$), the generator produces candidate questions ($\tilde{q}^m_i$) together with sequence-level confidence scores from beam decoding. This stage ensures that generated questions are not sampled from a generic prompt distribution but from a task-aware conditional distribution aligned with incremental learning requirements. The generated candidates are then refined by a multi-level filtering stack. First, a surface-form filter removes malformed questions using simple but effective constraints (minimum token length and interrogative ending). Second, duplicate questions are removed to increase diversity and reduce redundant signals. Third, the router from step $t-1$ (before it is updated at step t) predicts the question's task label from text alone; only questions whose predicted label matches the intended previous task are retained, which substantially reduces cross-task contamination. After task-consistency screening, retained questions are ranked by normalized confidence (length-normalized sequence log-likelihood proxy), and only the top ($\rho\%$) are kept, yielding a high-confidence subset ($\mathcal{Q}^m_{\text{filtered}}$). Finally, each retained question is paired with its image and passed to the VQA expert to produce short answers, using the prompt as described in Appendix H, forming QA triplets ($(\mathbf{v}_i, \tilde{q}^m_i, \tilde{a}^m_i)$). Overall, this design couples task-aware prompting, semantic consistency checking, and confidence-based ranking, so generated samples are both high quality and task-faithful.

E.1 CLIP-BASED ALIGNMENT ANALYSIS.

To quantify image–question alignment, we compute CLIP cosine similarity using the pretrained `CLIP ViT-B/32` model. For each sample, we extract image and text embeddings, ℓ_2-normalize them, and compute s_i:

$$s_i = \left\langle \frac{f_{\text{img}}(I_i)}{\|f_{\text{img}}(I_i)\|_2}, \frac{f_{\text{txt}}(Q_i)}{\|f_{\text{txt}}(Q_i)\|_2} \right\rangle. \tag{8}$$

We report per-task statistics over three settings: Real, Generated, and Shuffled Negative. In the Real setting, Q_i is the VQACL (Zhang et al., 2023) task questions. In the Generated setting, Q_i is the generated question. In the Shuffled Negative setting, for each task t, generated questions from task t are paired with images sampled from other tasks, while excluding any image appearing in task t (i.e., cross-task mismatched pairs).

For each task, we report the mean and standard deviation of $\{s_i\}$. Additionally, we define a task-specific threshold δ_t as the 10th percentile of the Real score distribution:

$$\delta_t = \text{Percentile}_{10}\left(\{s^{\text{real},t}_i\}\right), \tag{9}$$

and measure the proportion of samples above this threshold:

$$\text{Pct}^{(\cdot,t)}_{>\delta_t} = 100 \times \frac{1}{N_{(\cdot,t)}} \sum_{i=1}^{N_{(\cdot,t)}} \mathbb{1}\left[s^{(\cdot,t)}_i > \delta_t\right]. \tag{10}$$

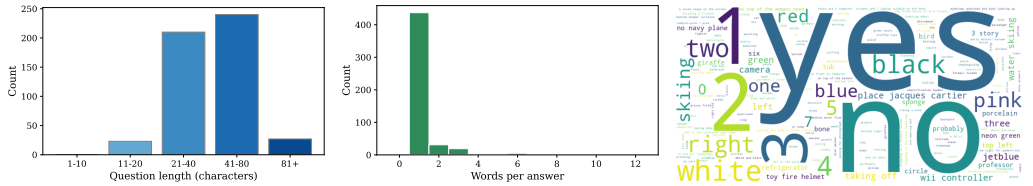
This provides a calibrated comparison of how often Generated and Shuffled Negative pairs exceed a low-tail reference level of Real data for each task.

Table 12 shows a consistent ordering across question types: Generated $>$ Real $\gg$ Shuffled Negative. First, generated questions achieve higher CLIP similarity than real questions in all 9 comparable tasks (excluding the last task, which is unavailable in generated data), with gains ranging from approximately $+0.009$ to $+0.021$. This suggests that generated questions are, on average, more aligned with visual content in the CLIP embedding space.

Second, shuffled negatives yield substantially lower scores (typically $\sim$0.17–0.19) than both real ($\sim$0.23–0.25) and generated ($\sim$0.25–0.27) pairs. This large margin indicates that the metric is sensitive to image–question correspondence rather than reflecting only dataset-level priors. The threshold-based analysis further supports this: using δ_t (the 10th percentile of each task's real distribution), generated pairs maintain high pass rates (95.2%–98.6%), whereas shuffled negatives remain low (16.1%–33.2%). Hence, most generated pairs lie above the low-tail real-quality boundary, while mismatched pairs rarely do.

Table 12: CLIP similarity statistics by question type. δ_t is defined as the 10th percentile of the Real distribution for each task.

Metric	Rec.	Loc.	Jud.	Com.	Cou.	Act.	Col.	Typ.	Sub.	Avg.
Real Ques.	0.24±0.03	0.23±0.02	0.24±0.03	0.24±0.03	0.24±0.03	0.23±0.03	0.23±0.03	0.23±0.03	0.23±0.03	0.24±0.03
Generated Ques.	0.25±0.03	0.25±0.02	0.26±0.03	0.26±0.03	0.26±0.03	0.25±0.02	0.24±0.03	0.25±0.03	0.25±0.03	**0.25±0.03**
Shuffled Neg. Ques.	0.17±0.02	0.18±0.02	0.18±0.02	0.17±0.02	0.17±0.02	0.18±0.02	0.17±0.02	0.16±0.02	0.17±0.02	0.17±0.02
δ_t (10th pct. Real)	0.19	0.19	0.20	0.19	0.19	0.19	0.19	0.19	0.19	0.19
%> δ_t Real Ques.	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0	90.0
%> δ_t Generated Ques.	96.6	97.6	98.0	98.6	97.8	96.7	95.2	97.3	95.4	**97.0**
%> δ_t Shuffled Neg. Ques.	17.6	33.2	16.1	18.6	17.5	28.8	21.9	16.8	20.1	21.2



(a) (b) (c)

Figure 6: Additional Overall statistics of our Compositional VQA dataset. (a) Question length in characters, (b) Answer word counts. (c) answer word cloud.

At the task level, `judge`, `commonsense`, and `count` show the highest generated means, while `color` remains the lowest among generated tasks. Interestingly, shuffled-negative pass rates are relatively higher for `location` and `action`, likely because these questions can remain partially plausible for many unrelated images. Overall, the results suggest that generated questions preserve meaningful image grounding and that the proposed evaluation protocol can clearly separate matched from deliberately mismatched image–question pairs.

F Additional Details of Benchmark

F.1 VQACL Benchmark

The VQACL benchmark is based on the VQA v2 dataset (Goyal et al., 2017). The VQA v2 contains human-annotated question and answer pairs over real-world images. The continual visual question answering (i.e., VQACL (Zhang et al., 2023)) benchmark is based on the VQA v2 benchmark's Karpathy splits of MSCOCO (Karpathy & Fei-Fei, 2015; Lin et al., 2014). Zhang et al. (2023) specifically designed ten linguistically driven tasks: *Recognition*, *Location*, *Judge*, *Commonsense*, *Count*, *Action*, *Color*, *Type*, *Subcategory*, and *Causal*. This design creates language-based hard task boundaries. However, images are shared between multiple tasks. The splits contain 131,478 (*Recognition*), 12,580 (*Location*), 160,179 (*Judge*), 25,211 (*Commonsense*), 62,156 (*Count*), 33,633 (*Action*), 50,872 (*Color*), 23,932 (*Type*), 31,594 (*Subcategory*), and 5,868 (*Causal*) train set questions. We use the task train-test splits provided by Huai et al. (2025).

F.2 COCO GQA Benchmark

The GQA (Hudson & Manning, 2019) dataset is based on compositional question answering. The compositional setup requires multi-step reasoning to answer a question. The dataset, however, is created using the Visual Genome scene graph and question engine. The Visual Genome shares many images with the COCO dataset, which was used in the VQACL benchmark. The VQACL test task questions are based on 5,000 COCO images. Based on those images, we filter GQA questions. A total of 41,981 questions from the GQA val set share the same images as the VQACL test tasks. We select all of these questions and construct the COCO-GQA compositional benchmark.

G Compositional VQA

Background. Existing benchmarks have notable limitations for compositional reasoning in VQA. For example, the VQACL (Zhang et al., 2023) benchmark does not include compositional questions,

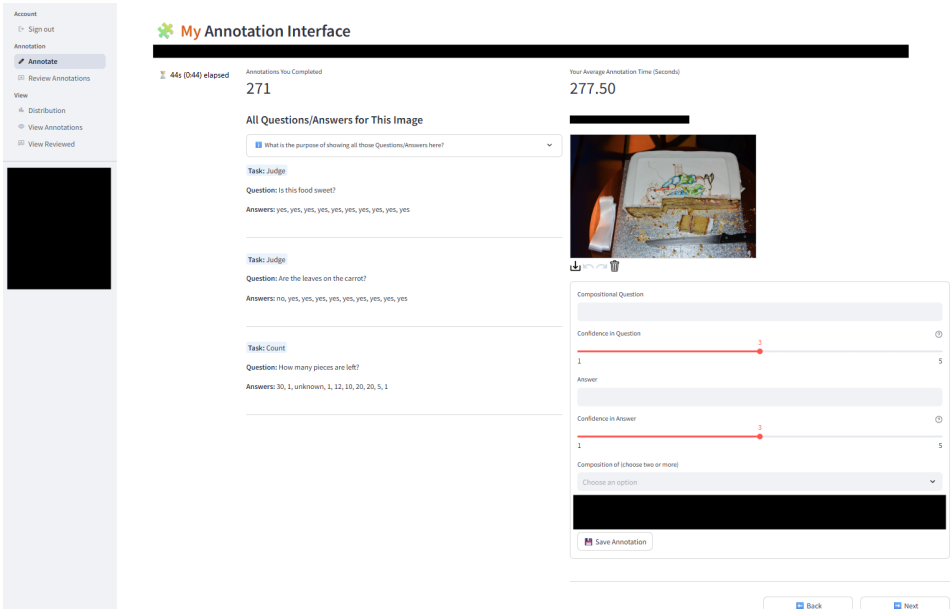


Figure 7: Sample Annotation Form. During the annotation process, annotators were presented with the corresponding images along with all human-asked questions associated with those images. This context was provided to help annotators reason more effectively when creating compositional questions.

while GQA (Hudson & Manning, 2019) is automatically generated (not human-annotated) and was not designed with continual learning in mind. Consequently, these datasets lack explicit tags identifying compositional tasks (i.e., task identities are not available). In contrast, compositional questions are more practical and reflective of real-world settings. In practice, a VQA learner acquires tasks sequentially, but during deployment, human users rarely restrict themselves to asking questions tied to a single task. Instead, it is natural for humans to ask compositional questions that combine multiple learned concepts. Due to the absence of benchmarks targeting compositionality, prior continual VQA methods have primarily focused on mitigating catastrophic forgetting. However, given the open-ended nature of the VQA task, compositional questions, which require multi-step visual reasoning, are both more practical and more reflective of real-world usage. As described in Section 4, we design our continual learning framework to solve such a practical application. To further test our method on human-annotated compositional questions, we create a benchmark, which in this paper we refer to as compositional VQA.

Benchmark Design. To create a composed test set, we selected all images from the VQACL test tasks. In total, the ten VQACL tasks share 5,000 images from MSCOCO. We divided these 5,000 images among sixteen students and collected a small subset of composed questions. During the annotation process, we also recorded the task identities to which each composed question belongs. Annotating task identities is important because our goal is to evaluate continual learners only on compositions of previously learned tasks. Figure 7 shows our sample annotation form. Figure 6 shows additional overall statistics of our compositional VQA dataset. We will make our test subset publicly available.

H Implementation Details

We instantiate our framework on two backbone MLLM architectures: BLIP-2 (Li et al., 2023) and LLaVA-7B. For BLIP-2, we use ViT-g/14 (Fang et al., 2023) as the frozen vision encoder and FlanT5-XL (Chung et al., 2024) as the language model. For LLaVA-7B, we use ViT-L/14 (Radford et al., 2021) as the frozen vision encoder and Vicuna (Chiang et al., 2023) as the language backbone. In BLIP-2, the multimodal Q-Former is configured with a fixed query size of 32, while for LLaVA, the projection embedding dimension is set to 64. Our experts are optimized in an alter-

nating manner, thus we set $\lambda_{\mathrm{VQG}} = \lambda_{\mathrm{VQA}} = 1$, giving equal importance to both objectives. The language-based router uses a single-layer LSTM encoder. During evaluation, we use the prompt template "Question: Short answer: " for BLIP-2 and "Answer the question using a single word or phrase" for LLaVA. The VQG retention ratio ρ is set to 50% (i.e., keep top 50%). For memory-based baselines, we fix the replay buffer size to 5000 samples to ensure fair comparison across methods. Publicly available results are used whenever possible.

H.1 BLIP-2 Backbone

For our implementation, we use LoRA with a scalar of 0.1, a dropout rate of 0.1, and a bottleneck dimension of 64, without applying layer normalization. The base LLM is Flan-T5-XL, configured with 5 beams, a length penalty of -1, deterministic decoding (nucleus sampling disabled), and a temperature of 1. The Q-Former is initialized with 32 queries, while the vision encoder employs the EVA-CLIP-G ViT model with an image resolution of 224 and FP16 precision. All parameters are frozen except the adapters within the Q-Former. Training is conducted using the AdamW optimizer (β ranging from 0.9 to 0.999) with a learning rate of 1e-5, weight decay of 0.05, a layer-wise learning rate decay factor of 1, and 1000 warmup steps. During evaluation, we use the prompt template *"Question: Short answer: "* for BLIP-2.

H.2 LLaVA Backbone

We train all experts atop the Vicuna-7B LLaVA backbone using a low-rank mixture-of-experts (MoE) adaptation. We initialize from Vicuna-7B with the LLaVA v1.5 multimodal projector and a CLIP ViT-L/14-336 vision tower, and attach task-specific low-rank adapters to the language layers. The backbone is initialized from LLaVA, while new adapter weights are warm-started from the previous expert. Training uses a maximum sequence length of 2048 tokens and a LoRA rank of 32 with a scaling factor of 64 for the expert adapters. We optimize with AdamW for one epoch with batch size 16 per GPU, learning rate 2×10^{-4}, cosine learning rate schedule, warmup ratio 0.03, and weight decay 0, using bfloat16 precision with gradient checkpointing enabled. During evaluation, we use the prompt template *"Answer the question using a single word or phrase"* for LLaVA.

H.3 Workstation Setup

The experiments were conducted on a workstation equipped with three NVIDIA A100 PCIe GPUs (40 GB each), totaling 120 GB of GPU memory.

I Limitations

Our method addresses both knowledge retention and compositional knowledge in the continual visual question answering setting. The current design focuses on compositions of previously learned tasks, which aligns well with the continual learning paradigm; however, handling future compositional questions (i.e., zero-shot compositions) remains a limitation and lies outside the scope of this work. In addition, the expert pool grows linearly with the number of tasks. Since each expert consists only of low-rank adapters, quantization-aware low-rank adaptation techniques could further reduce the storage and deployment footprint (Xu et al., 2024).

J Source Code

To ensure reproducibility, we include detailed implementation settings and benchmark construction procedures in this Appendix. Upon publication, we will release the complete codebase and both compositional splits (COCO-GQA and Compositional VQA).