Bisecle: Binding and Separation in Continual Learning for Video Language Understanding

Yue Tan
School of Computer Science
University of New South Wales
Sydney, Australia
yue.tan@unsw.edu.au

Xiaoqian Hu
School of Computer Science
University of New South Wales
Sydney, Australia
xiaoqian.hu@student.unsw.edu.au

Hao Xue
School of Computer Science and Engineering
University of New South Wales
Sydney, Australia
hao.xue1@unsw.edu.au

Celso De Melo
DEVCOM Army Research Laboratory
USA
celso.miguel.de.melo@gmail.com

Flora D. Salim *
School of Computer Science and Engineering
University of New South Wales
Sydney, Australia
flora.salim@unsw.edu.au

Abstract

Frontier vision-language models (VLMs) have made remarkable improvements in video understanding tasks. However, real-world videos typically exist as continuously evolving data streams (e.g., dynamic scenes captured by wearable glasses), necessitating models to continually adapt to shifting data distributions and novel scenarios. Considering the prohibitive computational costs of fine-tuning models on new tasks, usually, a small subset of parameters is updated while the bulk of the model remains frozen. This poses new challenges to existing continual learning frameworks in the context of large multimodal foundation models, i.e., catastrophic forgetting and update conflict. While the foundation models struggle with parameter-efficient continual learning, the hippocampus in the human brain has evolved highly efficient mechanisms for memory formation and consolidation. Inspired by the rapid **Bi**nding and pattern **se**paration mechanisms in the hippocampus, in this work, we propose **Bisecle** for video-language **c**ontinual **le**arning, where a multi-directional supervision module is used to capture more cross-modal relationships and a contrastive prompt learning scheme is designed to isolate task-specific knowledge to facilitate efficient memory storage. Binding and separation processes further strengthen the ability of VLMs to retain complex experiences, enabling robust and efficient continual learning in video understanding tasks. We perform a thorough evaluation of the proposed Bisecle, demonstrating its ability to mitigate forgetting and enhance cross-task generalization on several VideoQA benchmarks.

*Corresponding Author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

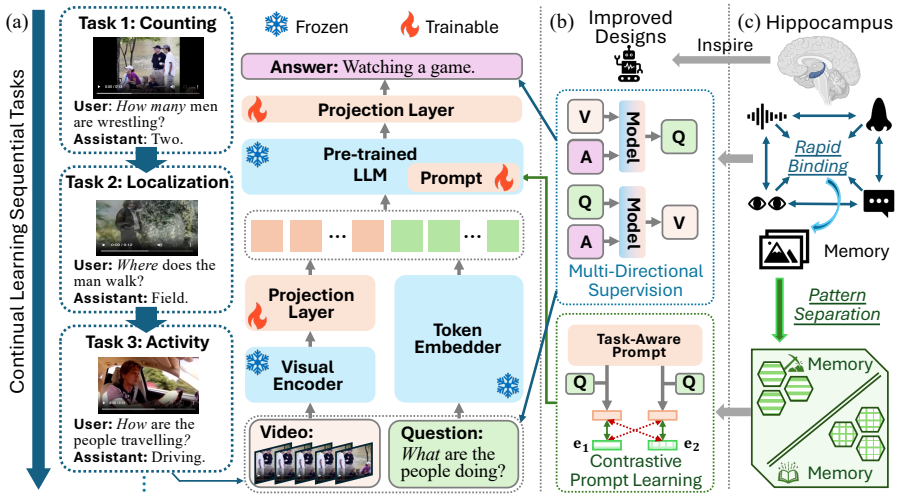


Figure 1: (a) The backbone of our continual learning framework for video understanding; (b) The improved designs proposed in this paper; (c) The motivations from a neurobiological perspective.

1 Introduction

Recent advances in large multimodal foundation models, such as vision-language models (VLMs) combining vision encoders with large language models (LLMs), have demonstrated remarkable capabilities in video understanding tasks like visual question answering (VideoQA) and video captioning [1, 2, 3]. However, in real-world applications, these models often face continuously evolving data streams, where the domain and data distribution exhibit inevitable shifts over time [4]. To address this challenge, continual learning has emerged as a critical research direction, enabling models to adapt to new tasks while retaining knowledge from previous ones [5, 6]. While traditional continual learning methods typically optimize the entire model, this approach becomes impractical in the context of large multimodal foundation models due to their massive scale and the prohibitive computational costs of fine-tuning the whole framework, including both the visual encoder and LLM [7, 8]. A more feasible alternative is to update only a small subset of pivot parameters while keeping the bulk of the model frozen [9, 10]. Figure 1(a) provides a sketch map of parameter-efficient continual learning for video understanding, where only the lightweight projection layers are learnable during the continual learning process to adapt to sequential tasks [9, 11].

Although updating only a small fraction of parameters can lead to great feasibility and training efficiency, this strategy also introduces significant challenges. ***Challenge 1*: catastrophic forgetting**. As the model is continuously trained on data from new tasks, the multimodal associations learned from previous tasks are gradually overwritten or lost. Due to the limited number of trainable parameters, the learnable modules struggle to maintain the knowledge for old tasks, making the forgetting problem more severe compared to full-model fine-tuning scenarios. ***Challenge 2*: update conflict**. Given the diversity and heterogeneity of video understanding tasks, they usually require the model parameters to induce incompatible gradient signals. When the trainable parameters are limited, these divergent adaptation needs inevitably lead to conflicts, as the scarce parameters are hard to simultaneously optimize for multiple, often incompatible objectives.

While large multimodal foundation models struggle with parameter-efficient continual learning, our brains have evolved highly efficient mechanisms to tackle similar challenges in real-world scenarios. Responsible for memory formation and consolidation, the hippocampus in the human brain employs several unique mechanisms to handle complex, dynamic information streams [12, 13]. To efficiently encode episodic memories, the hippocampus utilizes a **rapid binding** mechanism to link multimodal information dynamically into cohesive memory [14, 15]. As shown in the upper part of Figure 1(c), the visual, auditory, and contextual cues are integrated through synchronized neural activity, enabling the hippocampus to form multimodal memory traces. Meanwhile, to minimize interference between similar memories, the hippocampus employs **pattern separation** mechanism to generate distinct neural representations for overlapping inputs [16, 17]. As demonstrated in the lower part of Figure 1(c), this process ensures that memories of different events are stored in different

neural subspaces through sparse coding in the dentate gyrus. With efficient mechanisms like rapid binding and pattern separation, the hippocampus can achieve robust memory formation and retrieval using only 0.5% of total brain neurons. Considering the powerful memorizing capability of the hippocampus, a natural question arises: *Can these neurobiological principles inspire the design of parameter-efficient continual learning frameworks for video understanding?*

To answer the above question, in this paper, we propose a simple yet effective method, **Bisecle**, inspired by rapid **Bi**nding and pattern **se**paration mechanisms, for video-language **c**ontinual **le**arning (see Figure 2 for the icon). To be more specific, to address ***Challenge 1***, we design a multi-directional supervision module (as shown in the upper part of Figure 1(b)) to mimic the hippocampus's rapid binding capability. Specifically, apart from the task-specific training objective that takes videos and user questions as inputs and expected answers as outputs, we introduce auxiliary reconstruction tasks, including predicting questions from videos and answers and generating videos from questions and answers, to enforce bidirectional binding between modalities. This approach improves the semantic alignment between cross-modal task elements, thereby mitigating catastrophic forgetting. Moreover, to handle ***Challenge 2***, we propose a contrastive prompt learning module (as shown in the lower part of Figure 1(b)) inspired by the pattern separation mechanism. Concretely, we introduce task-specific task type embeddings to store information for different tasks and leverage them to optimize the learnable prompts using a contrastive loss. This loss enhances the agreement between reweighted prompts obtained by attending to task-specific questions and their corresponding learnable task-specific knowledge, which alleviates the update conflict in shared parameters. In this way, Bisecle effectively isolates task-specific knowledge and reduces the potential update conflict across different tasks during the continual learning process. To sum up, the contributions of this paper are three-fold:

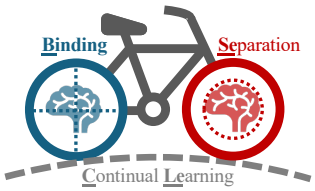


Figure 2: A "Bisecle" with two wheels (binding and separation) running on the road (a sequence of tasks).

- **Neurobiology-Inspired Problem Reframing.** We uniquely reframe this challenge through the lens of neurobiological mechanisms (i.e., rapid binding and pattern separation) to address catastrophic forgetting and update conflicts under strict parameter efficiency constraints.
- **Novel Methodology.** We propose a novel method, Bisecle, that integrates multi-directional supervision and contrastive prompt learning to enable robust and efficient continual learning in video understanding tasks with multi-modal LLMs.
- **Extensive Experiments.** We conduct extensive experiments to validate the effectiveness of Bisecle, demonstrating significant improvements in mitigating forgetting and enhancing cross-task generalization across three VideoQA benchmarks.

2 Related Works

Continual Learning for Large Language Models. Continual learning is a promising technique to extend large language models (LLMs) to evolving tasks and applications [18, 19, 20]. Existing methods to achieve continual learning on LLMs can be broadly categorized into three classes: continual pre-training [21, 22, 23, 24], continual instruction tuning [25, 26, 27], and continual alignment [28, 29, 30], each focusing on a specific stage in the deployment of LLMs [31]. Among them, continual instruction tuning most closely aligns with the objective of extending LLMs to video understanding tasks, since it enables the model to adapt to new tasks while retaining previously learned knowledge through task-specific instructions [32, 33, 26]. To address the catastrophic forgetting problem, Contunual-T0 [32] utilizes a memory buffer-based rehearsal mechanism to organize and replay the data of previous tasks. DynaInst [34] introduces dynamic instruction replay and local minima-inducing regularize to enhance the generalizability of models while preserving low computational cost. To enable parameter-efficiency continual learning with LLMs, Progressive Prompts [35] aims to freeze most parameters and only tunes the task-specific prompts for each task in the continual learning process. Jang et al. [36] propose to learn a small expert adapter on top of the LLM for each task and allocate a corresponding expert for each new-coming task. Despite their efficiency in natural language processing tasks, it is non-trivial to apply them to video understanding due to the unique challenges posed by multimodal data, such as the need for robust cross-modal alignment and the dynamic nature of video content.

Continual Learning for Video Understanding. To handle different video understanding tasks in a continual learning manner, recent studies explore existing continual learning techniques for video understanding models [4, 37, 38, 39, 40, 41, 42]. As one of the representation methods, CLOVE [37] utilizes a scene graph-based prompt mechanism for data replay in continual learning for videoQA. CL-VQA [38] models the intricate relationships between different modalities via multimodal decoupled prompts. DAM [43] introduces a dynamic merging mechanism for the parameters of adapters, which aims to handle the domain-incremental continual learning problem. In [44], a continual predictive learning approach is developed to learn a mixture world model via predictive experience replay. Tang et al. [6] propose a benchmark that systematically formulates the learning paradigm and provides a comprehensive evaluation of the existing methods. *Nevertheless, the above methods do not leverage LLMs for video understanding, thereby falling short in the comprehension and interpretation of semantic information.* To harness the powerful capability of LLMs for video understanding with evolving tasks, very recently, Cai et al. [9] propose the first LLM-based video understanding framework with continual learning. The proposed method, ColPro, incorporates various prompting techniques (i.e., question constraint, knowledge acquisition, and visual temporal awareness) to deal with the catastrophic forgetting issue. Different from ColPro, our proposed method Bisecle provides a simpler strategy that leverages a multi-directional supervision module and a contrastive prompt learning mechanism to strengthen the memorization process and reduce update conflicts, significantly enhancing the performance in continual video understanding.

3 Methodology

3.1 Problem Definition and Backbone Architecture

Problem Definition. In this paper, we focus on continual learning for video question answering (VideoQA), one of the most representative and essential tasks in video understanding. Consider a sequence of tasks indexed from 1 to T. Given the t-th task, we have its dataset $\mathcal{D}^t = \{d_1^t, \cdots, d_{n_t}^t\}$, where n_t is the number of data samples and each data sample $d_i^t =< V_i^t, Q_i^t, A_i^t >$ consists of three elements: video V_i^t, question Q_i^t, and answer A_i^t. The objective is to train a video-language model $f(V_i^t, Q_i^t) = A_i^t$ on the datasets of the sequence of tasks, and the model can achieve strong performance on both current (e.g., the T-th task) and previous (e.g., the $1, \cdots, T-1$-th tasks) tasks. In continual learning, different tasks should exhibit a certain level of diversity, reflected in differences in their data distributions [9, 37, 45]. A brief example of task evolving is demonstrated in the left part of Figure 1(a).

LLM-based Backbone Model. In this work, we consider a LLM-based video-language model serving as $f(V_i^t, Q_i^t) = A_i^t$ for video understanding tasks. Following Cai et al. [9], we employ the LLaMA-Adapter framework [11] with a ViT [46] visual encoder as our backbone model. As shown in the right part of Figure 1(a), the visual encoder takes the video V_i^t as its input, with a following projection layer to transfer it into a sequence of visual tokens $\mathbf{V}_i^t = \left[\mathbf{v}_1, \ldots, \mathbf{v}_{N_{i,v}^t}\right] \in \mathbb{R}^{N_{i,v}^t \times D}$, where $N_{i,v}^t$ is the number of frames in V_i^t, and D denotes the channel dimension for the extracted frame feature. Correspondingly, the question and answer tokens output by the pre-trained fixed tokenizer are denoted as $\mathbf{Q}_i^t = \left[\mathbf{q}_1, \ldots, \mathbf{q}_{N_{i,q}^t}\right] \in \mathbb{R}^{N_{i,q}^t \times D}$ and $\mathbf{A}_i^t = \left[\mathbf{a}_1, \ldots, \mathbf{a}_{N_{i,a}^t}\right] \in \mathbb{R}^{N_{i,a}^t \times D}$, respectively, where $N_{i,q}^t$ and $N_{i,a}^t$ denote the number of question tokens and answer tokens. Taking $\mathbf{V}_i^t$ and $\mathbf{Q}_i^t$ as the initial input, a frozen LLM with a trainable projection layer is expected to predict the answer tokens $\mathbf{A}_i^t$ in an autoregressive fashion. To fine-tune the learnable parameters of model f, a cross-entropy loss is employed, which can be written as:

$$\mathcal{L} = -\log P(A_i^t \mid V_i^t, Q_i^t) = -\sum_{k=0}^{N_{i,a}^t - 1} \log P\left(\mathbf{a}_{i,k+1}^t \mid \mathbf{V}_i^t, \mathbf{Q}_i^t, \mathbf{A}_{i,\leq k}^t\right), \tag{1}$$

where $\mathbf{A}_{i,\leq k}^t = [\mathbf{a}_{i,1}^t, \cdots, \mathbf{a}_{i,k}^t]$ includes the first k tokens of the answer sequence. Note that the modules with a large number of parameters (i.e., the LLM and visual encoder) are frozen, leaving only a small number of adaptive modules as learnable. This design ensures parameter-efficient training and adaptation throughout the continual learning process with sequential tasks.

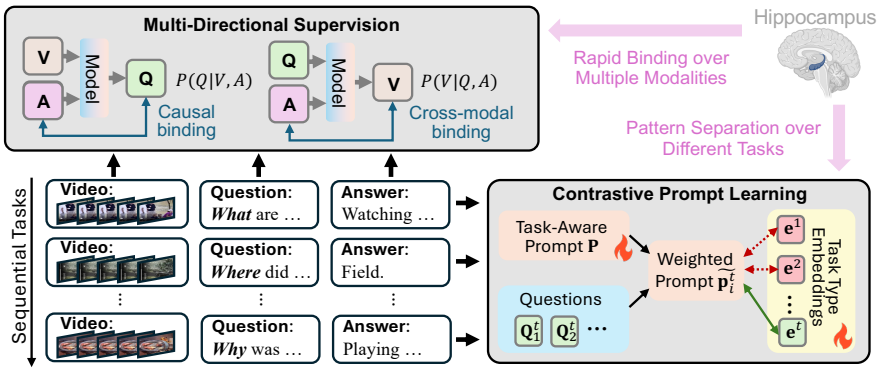


Figure 3: The sketch map to explain the two modules in Bisecle, multi-directional supervision and contrastive prompt learning.

3.2 Rapid Binding-Inspired Multi-Directional Supervision

Although the aforementioned backbone model can leverage the reasoning capabilities of LLMs while enabling efficient continual fine-tuning, like other continual learning systems, its performance is still impacted by the issue of **catastrophic forgetting** [9, 47, 48]. Specifically, the model may lose previously learned knowledge, such as the ability to focus on key frames and perform question-specific semantic reasoning in older video understanding tasks, when trained on new tasks, leading to performance degradation. In the context of frozen LLM and visual encoder, this forgetting phenomenon becomes even more severe. This is because the neurons responsible for visual and language understanding are fixed, making it difficult for the model to learn knowledge for scenario-specific understanding required for new tasks. Due to the single training mode where the model is only trained to predict answers given videos and questions, the learnable adapters struggle to capture deeper task understanding, such as the underlying semantic reasoning required for complex video understanding tasks, as well as the intricate cross-modal relationships between visual and textual information. This limitation hinders their ability to adapt effectively across different new tasks, particularly those requiring nuanced multimodal interactions.

Neurobiological Motivation - Rapid Binding. Given the similar challenge of memory and knowledge establishment, how does the human brain address this issue? The hippocampus, a critical region for memory formation, employs a rapid binding mechanism to dynamically link multimodal information, such as visual, auditory, and contextual cues, into cohesive episodic memories. This process is facilitated by synchronized neural activity, particularly theta-gamma oscillations, which enable efficient encoding and retrieval of complex associations. Importantly, this binding is **multi-directional**: given a partial cue (e.g., a visual scene), the hippocampus can reconstruct the associated context (e.g., the corresponding event or emotion), and **vice versa**, ensuring robust memory retention even under partial or noisy inputs. By deepening the understanding of multimodal information through these dynamic associations, the hippocampus strengthens memory traces and enhances the ability of brains to retain and recall complex experiences.

Multi-Directional Supervision. Inspired by the rapid binding mechanism of the hippocampus [15, 49, 50, 51], we propose a multi-directional supervision module for continual learning of LLM-based video-language models. Considering the VideoQA task, an essential data sample, corresponding to a "memory" for video-language models, is composed of video V_i^t, question Q_i^t, and answer A_i^t. The conventional learning objective is to maximize $\log P(A_i^t \mid V_i^t, Q_i^t)$, i.e., the likelihood of generating the correct answer A_i^t given the video V_i^t and question Q_i^t. Despite its alignment with the actual downstream task, the current objective is single-directional. That is to say, the model only learns the unidirectional connection from videos and questions to answers, while ignoring the potential for bidirectional or multi-directional associations, such as the connections from V_i^t, A_i^t to Q_i^t and which from Q_i^t, A_i^t to V_i^t. Such a single-directional supervision may limit the ability of the model to capture deeper causal and cross-modal relationships, thereby preventing it from forming a comprehensive understanding of the current training task. As a result, with insufficient learnable parameters, the model becomes more vulnerable to suffering from catastrophic forgetting.

To address the limitations of single-directional supervision, in Bisecle, we produce a multi-directional supervision module that incorporates multiple directions of learning objectives among video V_i^t, question Q_i^t, and answer A_i^t. Concretely, to explicitly bind the causal relationship between question Q_i^t and its corresponding video V_i^t and answer A_i^t, we introduce a question prediction task as auxiliary supervision. Formally, the learning objective function can be written as:

$$\mathcal{L}_Q = -\log P(Q_i^t \mid V_i^t, A_i^t) = -\sum_{k=0}^{N_{i,q}^t-1} \log P\left(\mathbf{q}_{i,k+1}^t \mid \mathbf{V}_i^t, \mathbf{A}_i^t, \mathbf{Q}_{i,\leq k}^t\right), \tag{2}$$

where $\mathbf{Q}_{i,\leq k}^t = [\mathbf{q}_{i,1}^t, \cdots, \mathbf{q}_{i,k}^t]$ includes the first k tokens of the question sequence. Through the question prediction loss $\mathcal{L}_Q$, the model can capture more causal knowledge about the specific VideoQA task. For example, when given a video of a car accident and the answer "brake failure", the model can infer the question "what caused the accident?", which enables the model to establish a deeper understanding of the causal relationship between the question and the answer.

Apart from the question prediction task, Bisecle also incorporates a video prediction task to provide extra supervision signals. Through predicting the video sequence based on the question and answer, the video-language model can enhance its ability to capture cross-modal relationships as well as temporal dynamics, leading to a more robust understanding of the task content. However, it is challenging to directly predict the visual tokens as they are out of the discrete token space of language models. Inspired by [52], we adopt an alternative strategy that maximizes the mutual information between the input visual tokens of frames and the output feature of LLMs. Specifically, the learning objective function of our video prediction task can be written as:

$$\begin{aligned}\mathcal{L}_V = -\log P(V_i^t \mid Q_i^t, A_i^t) &= -\sum_{k=0}^{N_{i,v}^t-1} \log P\left(\mathbf{v}_{i,k+1}^t \mid \mathbf{Q}_i^t, \mathbf{A}_i^t, \mathbf{V}_{i,\leq k}^t\right) \\ &= -\sum_{k=0}^{N_{i,v}^t-1} \log \frac{\exp\left({\mathbf{v}_{i,k+1}^t}^\top \mathbf{h}_{i,k}^t\right)}{\sum_{j=1}^{N_{i,v}^t} \exp\left({\mathbf{v}_{i,j}^t}^\top \mathbf{h}_{i,k}^t\right)},\end{aligned} \tag{3}$$

where $\mathbf{h}_{i,k}^t$ is the output token representation of LLMs before the start of visual tokens. The video prediction loss encourages the model to predict the sequence of video frames based on the preceding frames, thereby enhancing its ability to capture temporal dependencies and improve cross-modal video understanding within the current training task in the continual learning process.

3.3 Pattern Separation-Inspired Contrastive Prompt Learning

Apart from the above challenge, another critical issue is the **update conflict** of the learnable parameters. To be concrete, when multiple tasks compete for updates to the same set of parameters, the model may prioritize new tasks at the expense of previously learned knowledge. Especially when the learnable parameters are scarce, this conflict becomes more pronounced, leading to significant interference between tasks. While existing approaches employ task-specific prompts [9, 53] to mitigate this issue, they require precise matching between test-time questions and training tasks to select the corresponding prompts. This matching process proves particularly challenging in open-world QA scenarios, where additional knowledge is often needed to identify the correct task association. Moreover, the substantial divergence between task-specific prompts may lead to overfitting to individual tasks, consequently degrading model performance. Therefore, how to alleviate the update conflict issue without introducing task-specific parameters remains a significant challenge.

Neurobiological Motivation - Pattern Separation. From the perspective of neurobiology, the memory encoding mechanism of our brains can provide a promising solution to deal with the update conflict issue. The dentate gyrus, a critical region in the hippocampus for higher-order cognitive functions, employs a pattern separation mechanism to encode distinct representations for overlapping inputs. This process is facilitated by sparse coding and lateral inhibition, which ensure that similar but non-identical memories are stored in **non-overlapping neural subspaces**. Notably, this mechanism allows the brain to differentiate between similar experiences while preserving the unique details of

each knowledge. By isolating task-specific knowledge through pattern separation, the hippocampus can efficiently retain and recall complex experiences without interference.

Contrastive Prompt Learning. To enable the model to learn task-specific knowledge while maintaining model generalization ability and parameter efficiency, we introduce task-aware learnable prompts that shared across all training tasks and then introduce a contrastive prompt learning strategy to optimize the prompts with task-specific restrictions. Formally, the task-aware learnable prompts are attached to multiple transformer layers in the LLM as adaptive task adapters, which can be denoted as a prompt matrix $\mathbf{P} \in \mathbb{R}^{(N_p \times L_p) \times D}$, where N_p is the number of learnable prompt tokens at each layer, L_p is the number of transformer layers where prompts are injected, and D is the feature dimension of LLMs. Here, N_p and L_p are hyper-parameters to define the size of prompts. At the corresponding layer, the N_p learnable prompt embeddings are concatenated with the question embeddings.

While shared prompts help preserve generalization ability across diverse and open-set downstream tasks during testing, they inevitably suffer from knowledge interference among different training tasks during prompt parameter updates. To mitigate this issue, we draw inspiration from the pattern separation mechanism and propose a contrastive prompt learning strategy. This approach regularizes learnable prompts with task-specific embeddings, explicitly strengthening their associations with task-specific knowledge. Specifically, we allocate a task type embedding $\mathbf{e}^t \in \mathbb{R}^D$ for each task t, serving as the non-overlapping neural subspace to store the knowledge for the specific task. We dynamically reweight the learnable prompts based on specific training questions and employ a contrastive loss to enforce mutual agreement between each reweighted prompt and its corresponding task type embedding. This approach enables task-aware prompts to adapt to both the specific tasks and questions through explicit alignment with task-related knowledge, thereby mitigating potential conflicts in these bottleneck parameters.

In formal, given a question token matrix $\mathbf{Q}_i^t$ and the task-aware prompt matrix $\mathbf{P}$, the reweighted prompt vector can be calculated by $\widetilde{\mathbf{p}}_i^t \in \mathbb{R}^D = \left(\mathbf{q}_i^t \cdot \mathbf{P}^\top\right) \cdot \mathbf{P}$, where $\mathbf{q}_i^t \in \mathbb{R}^D$ is the averaged question representation obtained by mean-pooling the token embeddings of $\mathbf{Q}_i^t$ along the sequence dimension. In this way, $\widetilde{\mathbf{p}}_i^t$ effectively encodes task-specific knowledge with the awareness of the current input context. Notably, more complicated mechanisms (e.g., attention across multi-layer outputs) can be applied to calculating $\widetilde{\mathbf{p}}_i^t$, but we empirically found that our lightweight reweighting achieves comparable performance with reduced computational overhead. Once we obtain $\widetilde{\mathbf{p}}_i^t$, the contrastive loss across $\widetilde{\mathbf{p}}_i^t$ and $\mathbf{e}^t$ can be calculated by:

$$\mathcal{L}_P = -\log \frac{\exp\left(\widetilde{\mathbf{p}}_i^t \cdot \mathbf{e}^t / \tau\right)}{\sum_{t' \in \mathcal{A}(t)} \exp\left(\widetilde{\mathbf{p}}_i^t \cdot \mathbf{e}^{t'} / \tau\right)}, \tag{4}$$

where $\mathcal{A}(t) = \{0, \cdots, t\}$ is the set denoting the index of previous and current tasks from 0 to t, and τ is the temperature parameter that can adjust the tolerance for feature difference. Through this contrastive mechanism, learnable prompts are discriminately aligned with the task-specific embeddings, mitigating the problem of inter-task interference and further alleviating update conflicts during continual learning. In this mechanism, the task type embedding matrix $\mathbf{e}^t$ functions analogously to the dentate gyrus in human memory systems, strategically partitioning different task representations into distinct latent regions. With a trade-off hyper-parameter γ, the overall training loss of Bisecle can be written as $\mathcal{L} = \mathcal{L}_A + \mathcal{L}_Q + \mathcal{L}_V + \gamma\mathcal{L}_P$.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct experiments on three VideoQA datasets, i.e., NExT-QA [54], DramaQA [55], and STAR [56]. For NExT-QA, we split questions into eight task types (e.g., causal why/how, temporal what/when, and descriptive where/how many). Following prior work [9], we adopt the task order <TP, CW, DC, TC, DL, DO, TN, CH>. For DramaQA, we partition questions into five types and use the order with maximum forgetting, i.e., <What, Who, Where, How, Why>. For STAR, we follow its reasoning tasks <Interaction, Sequence, Prediction, Feasibility> to evaluate situational understanding in the continual learning scenarios. More details are in Appendix A.1.1.

Table 1: Comparison with state-of-the-art methods on NExT-QA, DramaQA, and STAR datasets. **Bold** and underline represent the best and runner-up results in each column.

Method	NExT-QA		DramaQA		STAR	
	Acc (↑)	Fog (↓)	Acc (↑)	Fog (↓)	Acc (↑)	Fog (↓)
Backbone (LLaMA-Adapter)	46.58	13.83	60.99	24.39	46.89	11.54
L2P [53]	48.82	12.25	62.50	20.67	48.25	10.82
DualPrompt [57]	50.62	11.74	65.89	17.93	49.73	10.15
LAE [58]	49.38	11.47	65.82	17.35	49.15	9.87
DAM [43]	53.88	9.99	67.37	15.19	50.64	8.92
ProgPrompt [35]	53.95	10.69	67.92	14.95	<u>51.07</u>	8.75
ColPro [9]	<u>55.14</u>	<u>7.43</u>	<u>71.24</u>	<u>12.64</u>	48.67	<u>8.13</u>
Bisecle (ours)	**62.37**	**5.34**	**71.49**	**10.37**	**52.16**	**7.60**

Baselines. We compare Bisecle to 6 representative methods for visual/video continual learning, including L2P [53], DualPrompt [57], LAE [58], DAM [43], ProgPrompt [35], and ColPro [9]. Among them, ColPro also employs adapter [11] as its backbone, making it a direct counterpart to our method for a fair comparison. More details for baselines can be found in Appendix A.1.2.

Evaluation Metrics. We evaluate the performance of baselines and Bisecle with two commonly-used metrics [9, 38, 53]. To evaluate video understanding performance, we adopt the standard metric of average final accuracy (Avg. Acc) over T tasks for multiple-choice question answering. To evaluate their continual learning capability, we employ average forgetting (Avg. Fog) to quantify catastrophic forgetting, where a smaller value indicates better preservation of previously learned knowledge.

Implementation Details. We use LLaMA-Adapter [11] as our backbone model, following [9]. We use the pre-trained LLaMA-2-7B [59] as the LLM and ViT-L/14 [60, 61] as the visual encoder, both of which are fixed during the continual learning process. All models are trained for five epochs with a batch size of 32 on all datasets. The number of adapter layers is set to 32, the adapter length is 10, and the weight decay is 0.14. We conduct all experiments on two NVIDIA H100 GPUs. Detailed experimental settings can be found in Appendix A.

4.2 Experimental Results

Performance Comparison. The performance comparison on three benchmark datasets is demonstrated in Table 1, from which we have the following observations. ❶ Our method establishes new state-of-the-art results across all datasets, surpassing the backbone model in both accuracy (+15.79% on NExT-QA) and forgetting reduction (8.49% lower Fog on NExT-QA). The superior performance demonstrates the powerful capability to handle continual learning challenges and tackle catastrophic forgetting issues. ❷ The poor performance of the backbone (adapter) highlights its vulnerability to sequential learning, with forgetting rates up to 24.39% on DramaQA, underscoring the need for dedicated continual learning mechanisms for LLM-based video understanding models. ❸ While ColPro employs a more complicated prompt learning mechanism based on the same backbone, our approach achieves superior performance (↑2.35% Acc, ↓2.27% Fog on DramaQA) through lightweight multi-directional supervision and contrastive prompt learning mechanisms.

Ablation Studies. To investigate the contribution of each component in our method, we compare Bisecle with different variants with different removed loss terms. According to the results in Table 2, we have the below findings. ❶ The complete version of Bisecle utilizing all loss components demonstrates the best performance across all three datasets. This indicates synergistic effects between the different mechanisms (i.e., multi-directional supervision and contrastive prompt learning), where joint optimization maximizes model capability while minimizing catastrophic forgetting and alleviating update conflict. ❷ In most cases, all loss terms bring positive effects to the performance, indicating the effectiveness of the proposed mechanisms. ❸ Among the three loss terms, $\mathcal{L}_Q$ provides the most significant performance improvement, demonstrating that strengthening the causal relationship between questions and answers is critical for mitigating catastrophic forgetting.

Data Efficiency. To verify the performance of Bisecle under data-scarce scenarios, we conduct this experiment under varying training data sizes, as shown in Figure 4. The results demonstrate three key

Table 2: Ablation studies on the effects of different loss components across three datasets.

Components			NExT-QA		DramaQA		STAR	
$\mathcal{L}_Q$	$\mathcal{L}_V$	$\mathcal{L}_P$	Acc (↑)	Fog (↓)	Acc (↑)	Fog (↓)	Acc (↑)	Fog (↓)
✗	✗	✗	46.58	13.83	60.99	24.39	46.89	11.54
✓	✗	✗	61.13	6.63	71.40	10.90	49.71	9.43
✗	✓	✗	53.94	9.61	67.39	17.15	49.48	8.86
✗	✗	✓	55.82	8.93	64.74	19.01	47.93	9.19
✓	✓	✗	59.78	6.58	70.25	12.65	48.50	9.24
✗	✓	✓	51.95	11.73	67.84	14.10	49.68	7.97
✓	✗	✓	61.38	7.20	68.47	14.63	51.96	8.43
✓	✓	✓	**62.37**	**5.34**	**71.49**	**10.37**	**52.16**	**7.60**

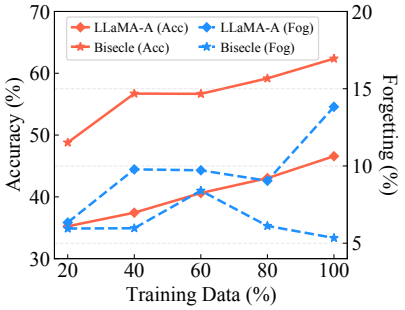


Figure 4: Performance on different sizes of training datasets.

Table 3: Performance on various sizes of LLMs (i.e., LLaMA-3.2-1B, LLaMA-2-7B, and LLaMA-2-13B).

Method	LLaMA	# of L. Par.	Acc (↑)	Fog (↓)
Backbone	1B	1,673,890	20.39	0.10
	7B	4,499,456	52.81	4.69
	13B	6,034,560	57.98	4.52
Bisecle	1B	1,777,920	21.17	0.00
	7B	4,540,416	53.45	2.60
	13B	6,085,760	59.43	3.03

findings: ❶ Our method Bisecle consistently outperforms the backbone (i.e., Adapter) across two metrics, indicating its strong robustness when training data is limited. ❷ Bisecle exhibits remarkable resistance to forgetting when sufficient training data is available, indicating the effectiveness of its key mechanisms, i.e., multi-directional supervision and contrastive prompt learning. ❸ Bisecle can achieve superior performance even in low-resource settings, indicating its data efficiency.

Robustness with Different LLMs. To validate the adaptation capability of Bisecle with different LLMs, we conduct experiments with LLaMA variants (1B/7B/13B parameters), and the results are shown in Table 3. For the sake of time, we only run experiments on three typical question types of NExT-QA dataset. We have these key observations emerge: ❶ Bisecle achieves accuracy improvements across all model sizes, demonstrating its robustness to LLM backbones. ❷ The forgetting rate is significantly reduced by Bisecle, especially for mid-scale models. This suggests our neurobiologically-inspired designs effectively mitigate catastrophic forgetting regardless of model scale. ❸ Despite adding only a few extra learnable parameters (104k–51k), Bisecle delivers disproportionate benefits.

Table 4: Performance with different numbers of prompt layers.

# P. Layer	Acc (↑)	Fog (↓)
8	53.86	10.13
16	58.18	7.52
24	57.30	7.99
32	62.37	5.34

Sensitivity to the Numbers of Layers with Prompt Injection. To study the impact of different numbers of learnable prompts, we vary the numbers of layers with prompt injection (# P. Layer) from 8 to 32, with the results shown in Table 4. The results demonstrate a clear scaling trend: increasing the number of learnable prompt layers usually improves both accuracy and forgetting resistance. This indicates that task-specific parameters play dual critical roles in continual learning with video-language models.

Table 5: Performance of different contrastive prompt learning manners.

Method	Acc (↑)	Fog (↓)
Variant 1	58.96	6.56
Variant 2	59.96	7.58
Bisecle (ours)	**62.37**	**5.34**

Performance of Varying Contrastive Prompt Learning Manners. Table 5 shows the performance of using different contrastive prompt learning manners on NExT-QA dataset. In variant 1, task type embeddings are computed as the mean of question tokens within the same class rather than as learnable embeddings. In variant 2, task type embeddings are directly used as the contrastive components without the prompt reweighting procedure. It can be observed that by defining the task type embedding as learnable variables, VLMs can better enhance performance on various tasks.

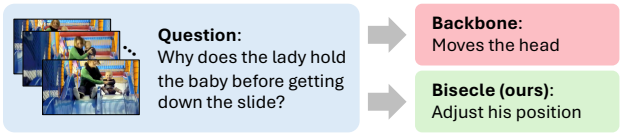


Figure 5: Answers by Backbone and Bisecle.

Case Study. We further evaluate the effectiveness of Bisecle by examining failure cases (See Figure 5, more can be found in Appendix B.3) of the backbone model where our approach succeeds. While the backbone model fails to establish causal relationships between questions and answers, Bisecle successfully resolves the case, thanks to the causal understanding capability acquired from multi-directional supervision.

5 Conclusion

In this paper, we present Bisecle, a neurobiologically inspired framework for video-language continual learning that addresses critical challenges in adapting large VLMs to dynamic real-world scenarios. By emulating the binding and pattern separation mechanisms in human brain through multi-directional supervision and contrastive prompt learning, Bisecle effectively mitigates catastrophic forgetting while maintaining parameter efficiency. It not only advances the state of continual learning for multimodal systems but also provides new insights into biologically inspired AI design. Extensive evaluations on VideoQA benchmarks demonstrate the superior ability of our method to preserve past knowledge and generalize to novel tasks compared to existing approaches.

References

[1] Yunlong Tang, Jing Bi, Siting Xu, Luchuan Song, Susan Liang, Teng Wang, Daoan Zhang, Jie An, Jingyang Lin, Rongyi Zhu, et al. Video understanding with large language models: A survey. *arXiv preprint arXiv:2312.17432*, 2023.

[2] Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360, 2023.

[3] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12585–12602, 2024.

[4] Keval Doshi and Yasin Yilmaz. Rethinking video anomaly detection-a continual learning approach. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3961–3970, 2022.

[5] Andrés Villa, Juan León Alcázar, Motasem Alfarra, Kumail Alhamoud, Julio Hurtado, Fabian Caba Heilbron, Alvaro Soto, and Bernard Ghanem. Pivot: Prompting for video continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24214–24223, 2023.

[6] Tianqi Tang, Shohreh Deldari, Hao Xue, Celso De Melo, and Flora Salim. Vilco-bench: Video language continual learning benchmark. *Advances in Neural Information Processing Systems*, 37:70213–70229, 2024.

[7] 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.

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

[9] Chen Cai, Zheng Wang, Jianjun Gao, Wenyang Liu, Ye Lu, Runzhong Zhang, and Kim-Hui Yap. Empowering large language model for continual video question answering with collaborative prompting. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3921–3932, 2024.

[10] Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023.

[11] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023.

[12] Larry R Squire, Heike Schmolck, and Shauna M Stark. Impaired auditory recognition memory in amnesic patients with medial temporal lobe lesions. *Learning & Memory*, 8(5):252–256, 2001.

[13] Christian Schultz and Maren Engelhardt. Anatomy of the hippocampal formation. *Frontiers of neurology and neuroscience*, 34:6–17, 2014.

[14] Rosanna K Olsen, Sandra N Moses, Lily Riggs, and Jennifer D Ryan. The hippocampus supports multiple cognitive processes through relational binding and comparison. *Frontiers in human neuroscience*, 6:146, 2012.

[15] Daniel M Cer and Randall C O'Reilly. Neural mechanisms of binding in the hippocampus and neocortex: insights from computational models., 2006.

[16] Michael A Yassa and Craig EL Stark. Pattern separation in the hippocampus. *Trends in neurosciences*, 34(10):515–525, 2011.

[17] Jill K Leutgeb, Stefan Leutgeb, May-Britt Moser, and Edvard I Moser. Pattern separation in the dentate gyrus and ca3 of the hippocampus. *science*, 315(5814):961–966, 2007.

[18] Xiao Wang, Yuansen Zhang, Tianze Chen, Songyang Gao, Senjie Jin, Xianjun Yang, Zhiheng Xi, Rui Zheng, Yicheng Zou, Tao Gui, et al. Trace: A comprehensive benchmark for continual learning in large language models. *arXiv preprint arXiv:2310.06762*, 2023.

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

[20] Yutao Yang, Jie Zhou, Xuanwen Ding, Tianyu Huai, Shunyu Liu, Qin Chen, Yuan Xie, and Liang He. Recent advances of foundation language models-based continual learning: A survey. *ACM Computing Surveys*, 57(5):1–38, 2025.

[21] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8968–8975, 2020.

[22] Andrea Cossu, Antonio Carta, Lucia Passaro, Vincenzo Lomonaco, Tinne Tuytelaars, and Davide Bacciu. Continual pre-training mitigates forgetting in language and vision. *Neural Networks*, 179:106492, 2024.

[23] Kuan-Ying Lee, Yuanyi Zhong, and Yu-Xiong Wang. Do pre-trained models benefit equally in continual learning? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6485–6493, 2023.

[24] Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. Continual pre-training of language models. *arXiv preprint arXiv:2302.03241*, 2023.

[25] Jinghan He, Haiyun Guo, Ming Tang, and Jinqiao Wang. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*, 2023.

[26] Cheng Chen, Junchen Zhu, Xu Luo, Hengtao Shen, Jingkuan Song, and Lianli Gao. Coin: A benchmark of continual instruction tuning for multimodel large language models. *Advances in Neural Information Processing Systems*, 37:57817–57840, 2024.

[27] Meng Cao, Yuyang Liu, Yingfei Liu, Tiancai Wang, Jiahua Dong, Henghui Ding, Xiangyu Zhang, Ian Reid, and Xiaodan Liang. Continual llava: Continual instruction tuning in large vision-language models. *arXiv preprint arXiv:2411.02564*, 2024.

[28] Han Zhang, Lin Gui, Yuanzhao Zhai, Hui Wang, Yu Lei, and Ruifeng Xu. Copf: Continual learning human preference through optimal policy fitting. *CoRR*, 2023.

[29] Alane Suhr and Yoav Artzi. Continual learning for instruction following from realtime feedback. *Advances in Neural Information Processing Systems*, 36:32340–32359, 2023.

[30] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19148–19158, 2023.

[31] Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. Continual learning for large language models: A survey. *arXiv preprint arXiv:2402.01364*, 2024.

[32] Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6107–6122, 2022.

[33] Yifan Wang, Yafei Liu, Chufan Shi, Haoling Li, Chen Chen, Haonan Lu, and Yujiu Yang. Inscl: A data-efficient continual learning paradigm for fine-tuning large language models with instructions. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 663–677, 2024.

[34] Jisoo Mok, Jaeyoung Do, Sungjin Lee, Tara Taghavi, Seunghak Yu, and Sungroh Yoon. Large-scale lifelong learning of in-context instructions and how to tackle it. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12573–12589, 2023.

[35] Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. Progressive prompts: Continual learning for language models. In *The Eleventh International Conference on Learning Representations*, 2023.

[36] Joel Jang, Seungone Kim, Seonghyeon Ye, Doyoung Kim, Lajanugen Logeswaran, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Exploring the benefits of training expert language models over instruction tuning. In *International Conference on Machine Learning*, pages 14702–14729. PMLR, 2023.

[37] Stan Weixian Lei, Difei Gao, Jay Zhangjie Wu, Yuxuan Wang, Wei Liu, Mengmi Zhang, and Mike Zheng Shou. Symbolic replay: Scene graph as prompt for continual learning on vqa task. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1250–1259, 2023.

[38] Zi Qian, Xin Wang, Xuguang Duan, Pengda Qin, Yuhong Li, and Wenwu Zhu. Decouple before interact: Multi-modal prompt learning for continual visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2953–2962, 2023.

[39] Lama Alssum, Juan Leon Alcazar, Merey Ramazanova, Chen Zhao, and Bernard Ghanem. Just a glimpse: Rethinking temporal information for video continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2474–2483, 2023.

[40] Hemang Chawla, Arnav Varma, Elahe Arani, and Bahram Zonooz. Continual learning of unsupervised monocular depth from videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8419–8429, 2024.

[41] Giulia Castagnolo, Concetto Spampinato, Francesco Rundo, Daniela Giordano, and Simone Palazzo. A baseline on continual learning methods for video action recognition. In *2023 IEEE International Conference on Image Processing (ICIP)*, pages 3240–3244. IEEE, 2023.

[42] Jaeyoo Park, Minsoo Kang, and Bohyung Han. Class-incremental learning for action recognition in videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13698–13707, 2021.

[43] Feng Cheng, Ziyang Wang, Yi-Lin Sung, Yan-Bo Lin, Mohit Bansal, and Gedas Bertasius. Dam: Dynamic adapter merging for continual video qa learning. *arXiv preprint arXiv:2403.08755*, 2024.

[44] Geng Chen, Wendong Zhang, Han Lu, Siyu Gao, Yunbo Wang, Mingsheng Long, and Xiaokang Yang. Continual predictive learning from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10728–10737, 2022.

[45] 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*, pages 19102–19112, 2023.

[46] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022.

[47] Xilai Li, Yingbo Zhou, Tianfu Wu, Richard Socher, and Caiming Xiong. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. In *International conference on machine learning*, pages 3925–3934. PMLR, 2019.

[48] Wenpeng Hu, Zhou Lin, Bing Liu, Chongyang Tao, Zhengwei Tao Tao, Dongyan Zhao, Jinwen Ma, and Rui Yan. Overcoming catastrophic forgetting for continual learning via model adaptation. In *International conference on learning representations*, 2019.

[49] Ivilin Stoianov, Domenico Maisto, and Giovanni Pezzulo. The hippocampal formation as a hierarchical generative model supporting generative replay and continual learning. *Progress in Neurobiology*, 217:102329, 2022.

[50] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

[51] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

[52] Dohwan Ko, Ji Lee, Woo-Young Kang, Byungseok Roh, and Hyunwoo Kim. Large language models are temporal and causal reasoners for video question answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4300–4316, 2023.

[53] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022.

[54] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786, 2021.

[55] Seongho Choi, Kyoung-Woon On, Yu-Jung Heo, Ahjeong Seo, Youwon Jang, Minsu Lee, and Byoung-Tak Zhang. Dramaqa: Character-centered video story understanding with hierarchical qa. In *Proceedings of the aaai conference on artificial intelligence*, volume 35, pages 1166–1174, 2021.

[56] Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. Star: A benchmark for situated reasoning in real-world videos. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[57] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648. Springer, 2022.

[58] Qiankun Gao, Chen Zhao, Yifan Sun, Teng Xi, Gang Zhang, Bernard Ghanem, and Jian Zhang. A unified continual learning framework with general parameter-efficient tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11483–11493, 2023.

[59] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[60] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

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

[62] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction (Section 1) accurately summarize our contributions: (1) a neurobiology-inspired problem reframing, (2) a novel methodology for video understanding in continual learning, and (3) empirical validation on multiple benchmarks.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Limitations are explicitly stated in Appendix C.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Full experimental details (datasets, hyperparameters, random seeds) are in Appendix A. Code/data are provided in supplementary documents.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: Anonymized code and preprocessed data are publicly available. Original datasets are cited in Section 4.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: In Appendix C, we list hyperparameters (learning rates, batch sizes), and describe optimization method.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: Results report the mean of test accuracy and forgetting rate over all historical tasks, which is a typical measurement in continual learning paradigm. We do not report error bars due to the large computational costs required and limited computational resources.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Experiments used 2*H100 GPUs (500 GPU-hours total). Details are provided in Appendix A.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We confirm compliance with the NeurIPS Code of Ethics. No identifiable data or harmful applications are involved.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discussed the broader impact in Appendix C.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

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

11. **Safeguards**

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

    Answer: [NA]

    Justification: Our paper poses no such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All datasets are public datasets and well cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [NA]

    Justification: Our paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

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

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

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [NA]

    Justification: This research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Experimental Details

A.1 Detailed Setting of Table 1

A.1.1 Task and Data Setting

For NExT-QA and DramaQA datasets, we follow the continual learning setting in ColPro [9]. For NExT-QA, there are eight tasks including causal questions (CW, CH), temporal questions (TC, TN, TP), descriptive questions (DC, DL), and other types of questions (DO). For DramaQA, there are five tasks including *what*, *who*, *where*, *how*, and *why*. For STAR, there are four tasks corresponding to *interaction*, *sequence*, *prediction*, and *feasibility*. Table 6, Table 7, and Table 8 give several question examples for each task in NExT-QA, DramaQA, and STAR, respectively.

Table 6: Question examples of different tasks in NExT-QA.

Task Type	Question Examples
TP	What did the man in grey do before the plane took off? What did the man on the stage do before sitting?
CW	Why does the man have to throw the plane first in the middle of the video? Why did the man wear hat while riding the horse?
DC	How many children are in the video? How many people are cycling?
TC	What did the lady do when they reached the bush? What does the man in white do when he walks onto the stage?
DL	Where is the boy projecting his photos on? Where is the cat lying?
DO	What are the different colors of the balls floating in the pool? What is the possible relationship between the girl and the boy?
TN	How did the two women react after the woman in stripes released their hand? How did the dog react after the lady caress it?
CH	How did the boy in stripped open the book to see its contents? How did the child use his hands to show his excitement at the start?

Table 7: Question examples of different tasks in DramaQA.

Task Type	Question Examples
TW	What is being cooked in the pot? What kind of thing is to eat on the plate?
DO	Who enters the house with wearing a bag? Who is yelling?
DL	Where is Deogi looking while talking? Where did Sukyung put her hand?
CH	How does Jiya think will happen if Jiya doesn't come back with the lost money? How did Taejin start to play the game when Taejin is with chairman?
CW	Why was Deogi in the kitchen? Why did Haeyoung1 light a fire under the iron pot?

Table 8: Question examples of different tasks in STAR.

Task Type	Question Examples
Interaction	Which object was eaten by the person? Which object was put down by the person?
Sequence	Which object did the person eat after they put down the book? Which object did the person open after they sat at the table?
Prediction	What will the person do next? Which object would the person put down next after they take the bag?
Feasibility	What else is the person able to do with the dish? Which object is the person able to throw after walking through the doorway?

A.1.2 Details of Baseline Methods

L2P [53] reframes continual learning as a prompt-selection problem. It maintains a small pool of learnable prompts, each paired with a key, and keeps a large pre-trained transformer frozen. For each incoming sample, L2P computes a query from the input, retrieves the top-N matching prompts, prepends them to the input embeddings, and then updates only the prompt pool and a lightweight classifier via a combined cross-entropy and key-matching loss. This design decouples task-specific from shared knowledge, requires no rehearsal buffer or task IDs at test time, and consistently outperforms state-of-the-art methods across class-incremental, domain-incremental, and task-agnostic benchmarks.

DualPrompt [57] is a rehearsal-free continual learning framework that enables a frozen, pretrained vision transformer to learn a sequence of class-incremental tasks without storing any past data. It does so by introducing two small, complementary sets of learnable prompts (G-Prompts for capturing task-invariant "general" instructions and E-Prompts for encoding task-specific "expert" instructions), which are attached to selected multi-head self-attention layers. A simple matching mechanism retrieves the appropriate E-Prompt at test time, and the model is trained end-to-end with a combination of classification and prompt-matching losses.

LAE [58] introduces a unified approach for continual learning by leveraging parameter-efficient tuning methods, such as Adapter, LoRA, and Prefix, to adapt pre-trained models to new tasks efficiently. It consists of three key components: 1) learning with calibrated adaptation speeds to align different tuning methods, 2) accumulation of task-specific knowledge into an offline fine-tuning module via momentum updates, and 3) ensemble of online and offline modules during inference to balance performance across old and new tasks. This design ensures robust continual learning performance while minimizing catastrophic forgetting and computational overhead.

DAM [43] is a parameter-efficient method designed for continual video question answering. It mitigates catastrophic forgetting by training dataset-specific adapters while dynamically merging them during inference to handle unknown domains and enable knowledge sharing. DAM has demonstrated the effectiveness across diverse video and image tasks.

ProgPrompt [35] introduces a novel continual learning approach for language models by progressively learning and concatenating soft prompts for each new task while keeping the base model frozen. This method leverages task-specific prompts to prevent catastrophic forgetting and enables forward transfer by reusing knowledge from previous prompts. Additionally, it incorporates a residual MLP-based reparameterization technique to stabilize training and improve performance, achieving significant gains over existing methods in both few-shot and full-data settings.

ColPro [9] is a rehearsal-free continual learning framework that leverages a frozen large language model and three complementary prompting strategies: task-specific question constraint prompting, knowledge acquisition prompting, and visual temporal awareness prompting. These strategies work together to encode question context, multimodal knowledge, and temporal dynamics into prompts that steer the model to learn new tasks without overwriting prior knowledge.

A.1.3 Model Architecture

The whole model involves one LLM backbone which is the LLaMA-Adapter [11], one visual encoder which is ViT-L/14 [60, 61], a projection layer aligning the output latent representations of different modalities, and another projection layer as the final linear layer that maps the latent representations to logits over the vocabulary space.

For the LLM part, we have chosen LLaMA-7B as the backbone for most experiments except for the implementation to verify the robustness with different sizes of LLMs. We adopt LLaMA-Adapter as the parameter-efficient fine-tuning method. Concretely, instead of updating the full 7B parameters, LLaMA-Adapter freezes the pre-trained LLaMA and only learns the adaptation prompts with 1.2M parameters on top.

For the visual encoder, the input resolution is 224×224. For each input image, the encoder splits it into 14×14 non-overlapping patches, and each patch is flattened and linearly projected to $D = 1024$ dimensions. The visual encoder consists of 24 transformer encoder layers, each employing multi-head self-attention (16 heads) and a feedforward MLP with a hidden dimension of 4096 (GELU activation), followed by Layer Normalization (Pre-Norm) for stability.

A.1.4 Hyperparameters and Training Details

We use dataset-specific batch sizes together with AdamW across all tasks. In particular, for NExT-QA we set the batch size to 32, for DramaQA to 4, and for STAR to 16. All experiments employ the AdamW optimizer with a base learning rate of 0.09. Weight decay is 0.14 for NExT-QA and 0.10 for both DramaQA and STAR. Video inputs consist of 10 frames resized to 224×224, and token sequences are truncated or padded to 128 tokens for NExT-QA, 280 for DramaQA, and 150 for STAR. We train each model for 5 epochs (with 2 warm-up epochs) and fix the random seed to 0 for all tasks. Experiments are conducted on two NVIDIA H100 GPUs (94GB of memory per GPU). The GPU-hours for training is around 500 in total.

Table 9 shows the parameters of our contrastive learning setup, governing how question type representations are learned and how negative examples are weighted in the contrastive loss.

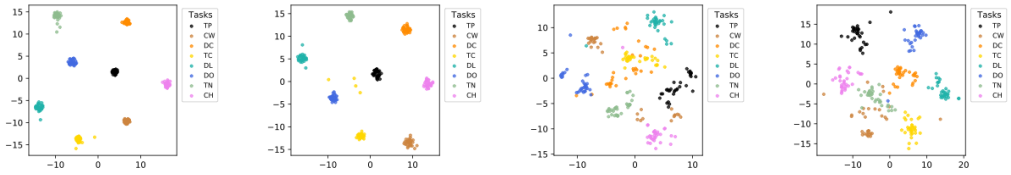
Table 9: Training Details of Contrastive Learning.

	NextQA	DramaQA	STAR
# Task Types	8	5	4
Task Type Embedding Size	[8, 4096]	[5, 4096]	[4, 4096]
Negative Temperature	1.28	1.25	1.25
Contrastive Loss Weight	0.15	0.10	0.10

B Additional Results

B.1 Visualization of Prompts across Tasks.

To learn about how the task type embeddings and weighted prompts of different tasks evolve along the continual learning process, we visualize the samples in NExT-QA test set by t-SNE [62]. In Figure 6, the points in different colors refer to representations after fine-tuning on a specific task for one epoch or four epochs. It suggests that, compared with weighted prompts, task type embeddings are more distinguishable in latent space, indicating that they retain more task-specific knowledge. Also, the distribution of weighted prompts is evolving along the task sequence in a more dynamic way, i.e., the positions of different clusters are changing across epochs, because new tasks generally refresh the understanding of the model on old tasks via task-aware prompt updates. This process subsequently affects the weighted prompts through contrastive learning mechanisms.



(a) Epoch 1 (b) Epoch 4 (c) Epoch 1 (d) Epoch 4

Figure 6: (a) and (b): t-SNE visualization of learnable *task type* embeddings of different tasks; (c) and (d): t-SNE visualization of *weighted prompts* of different tasks.

B.2 Performance on Varying Tasks Orders

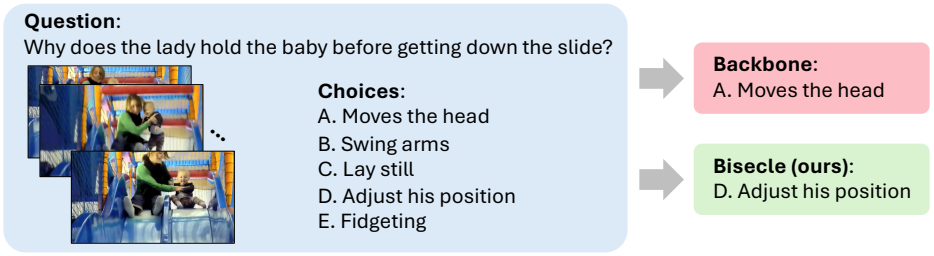
To investigate how the task order affects the continual learning performance, we evaluate the performance of LLaMA-Adapter and our method across different task orders. Table 10 suggests that the task order can influence both the test accuracy and forgetting rate, due to the various difficulty degrees of the tasks and their intrinsic correlation. It also shows that our method outperforms LLaMA-Adapter in both accuracy and forgetting rate in most cases, demonstrating the stability and robustness of our method to diverse continual learning task settings.

Table 10: Performance of LLaMA-Adapter and Bisecle (ours) across different task orders.

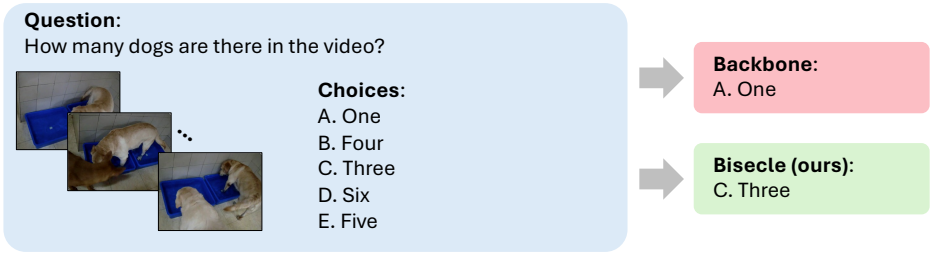
Task Order	Avg. Acc (↑)		Avg. Fog (↓)	
	LLaMA-A	Bisecle (ours)	LLaMA-A	Bisecle (ours)
<CH, DL, TP, TC, DC, DO, TN, CW>	56.79	63.09	5.55	2.87
<TP, TN, CH, TC, DL, DO, CW, DC>	57.68	57.98	5.89	7.16
<DO, CW, DC, CH, TP, TC, TN, DL>	55.63	57.95	7.15	8.93
<CW, DO, TN, DL, TC, TP, DC, CH>	52.65	62.25	12.85	5.70

B.3 Case Study

We present more cases on NExT-QA and STAR dataset in Figure 7 and Figure 8, respectively, where the backbone fails to give the right answer while our method succeeds. Each subfigure corresponds to a specific task in the learning sequence, helping the readers better learn about the video QA tasks we are working on.

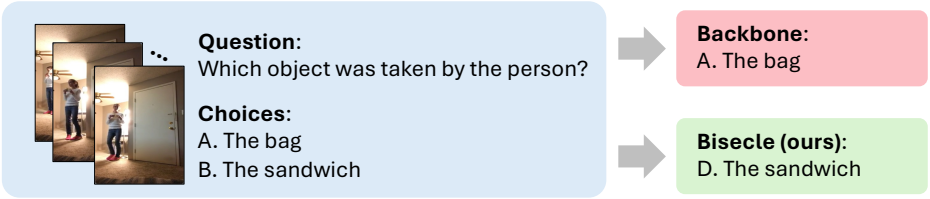


(a)

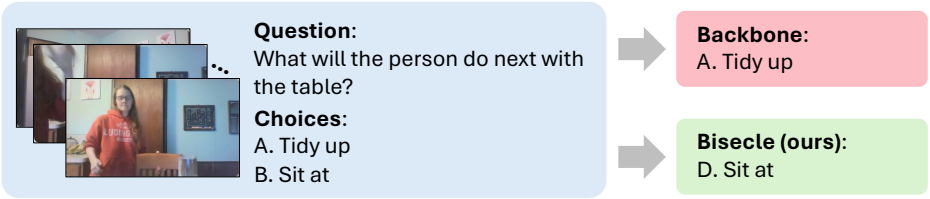


(b)

Figure 7: More cases on NExT-QA.



(a)



(b)

Figure 8: More cases on STAR.

B.4 Performance of Gemini and GPT

To investigate how frontier multimodal LLMs perform on the tasks in our continual learning sequence, we conduct a series of experiments based on Gemini and GPT family, i.e., Gemini 2.0 Flash, GPT-4o-mini, and GPT-4o. *It is worthwhile to note* that these frontier multimodal LLMs do not support parameter updates, and we can only use APIs to test their performance. Hence, the problems emphasized by this paper, i.e., catastrophic forgetting and update conflict, do not obviously exist for these frontier models and traditional continual learning metrics, e.g., the forgetting rate, cannot be measured appropriately. Although it is unfair for our method that experiences severe forgetting problems during the learning process to compare with these frontier models, this analysis still provides valuable insights into the capabilities of cutting-edge methods on such tasks, highlights their limitations, sheds light on open challenges, and guides subsequent improvements.

Performance of Gemini and GPT on the video-language tasks. As shown in Figure 9, we report the test accuracy of each task along the task sequence in NExT-QA dataset. It can be observed that Gemini 2.0 Flash achieves the highest accuracy in almost all tasks. Moreover, although our method experiences the negative impacts brought by continual learning setting, it achieves comparable performance in some tasks (CW, DC, TC, TN, CH) with GPT-4o-mini.

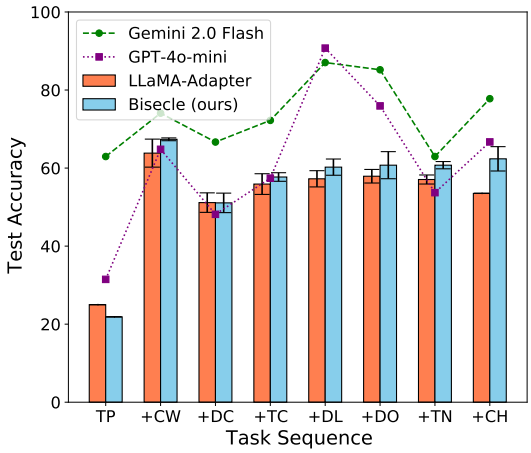


Figure 9: The performance of Gemini 2.0 flash, GPT-4o-mini, LLaMA-Adapter, and our method along the task sequence in the continual learning setting.

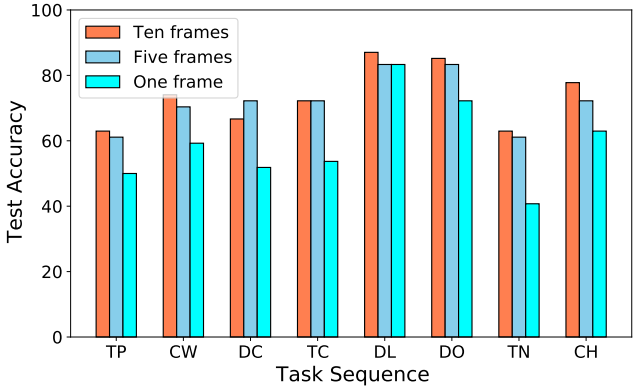


Figure 10: The performance of Gemini 2.0 Flash on different tasks with various numbers of input video frames. The experiments are conducted on NExT-QA dataset.

Weaknesses of Gemini and GPT. Despite the strengths of Gemini and GPT, the weaknesses of these frontier MLLMs still exist. *Firstly*, the zero-shot ability of the models highly depends on the number of input video frames. As shown in Figure 10, as the number of video frames decreases,

the performance on all tasks keeps dropping, demonstrating the dependency on sufficient or even redundant visual inputs. *Secondly*, the models have exhibited limited temporal reasoning ability, struggling with tasks that involve delayed causality and action/location sequencing capability. In Figure 11, it can be found that our method outperforms GPT-4o and GPT-4o-mini in task CW and DL, corresponding to "*why do*" and "*where is*", respectively. The former task requires the model to capture cause-and-effect relationships where the effect occurs at a time lag after the cause, rather than instantaneously, while the latter task requires the model to be equipped with sequencing ability, that is to acquire the temporal relationship of different actions and locations. Figure 12 gives a case study of the aforementioned issues faced by frontier models. Specifically, Figure 12(a) shows two failed cases related to delayed causality and Figure 12(b) shows them related to location sequencing.

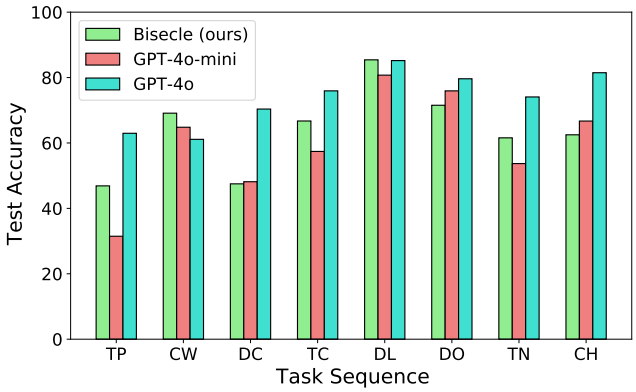
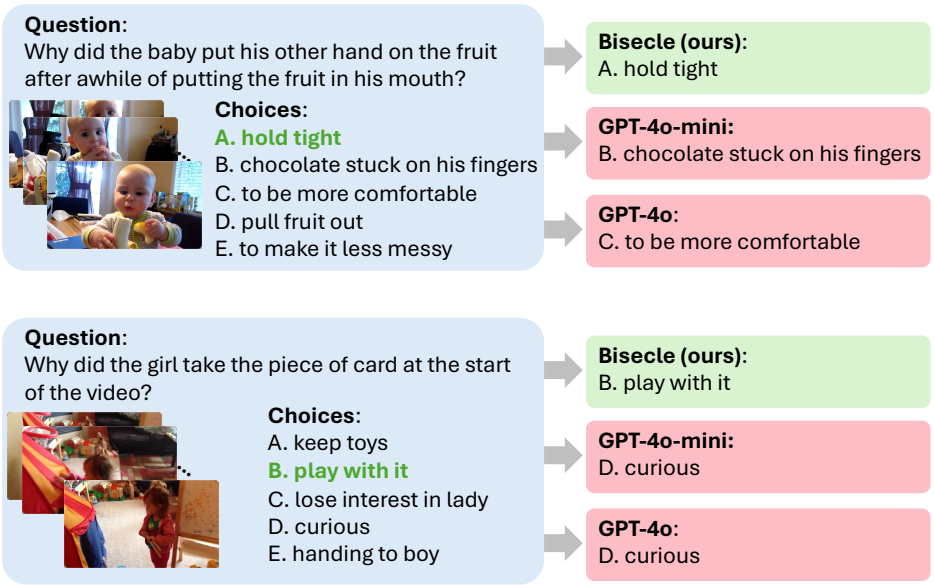
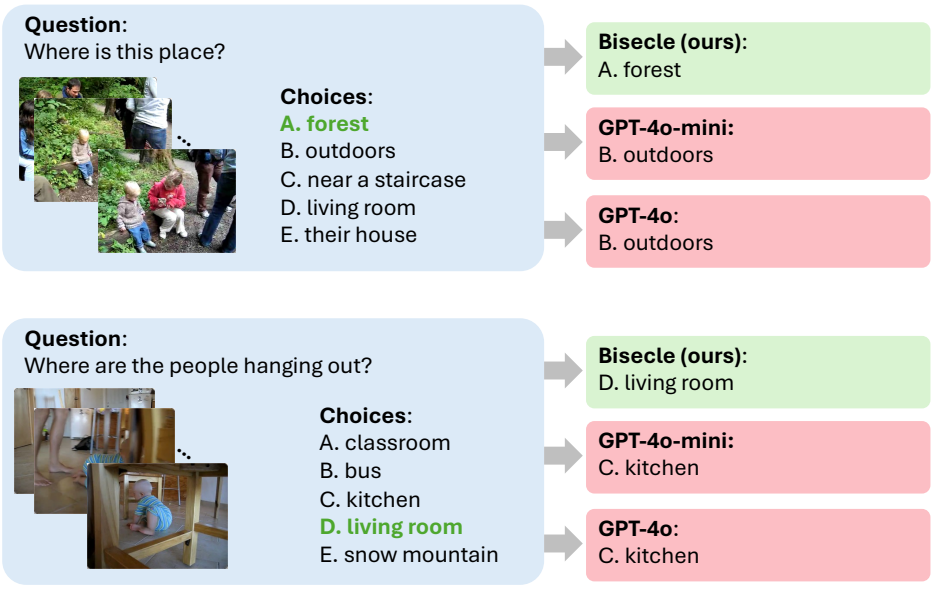


Figure 11: The performance of GPT-4o and GPT-4o-mini on the learning tasks of NExT-QA in our continual learning setting.



(a) Delayed causality



(b) Action/location sequencing

Figure 12: The weakness of frontier VLMs in dealing with continual learning tasks can be found as limited temporal reasoning ability, which makes them struggle with tasks that involve (a) delayed causality and (b) action/location sequencing.

B.5 Experiments on Varying LLM Backbones

To examine the generalizability of Bisecle, we further conducted extra experiments on more open-source backbone LLMs, including QWEN-7B and Gemma-7B, with the experimental results shown in Table 11. According to the results, we can see that Bisecle consistently leads to the optimal results compared to the baseline model (i.e., LLaMA-Adapter). This observation illustrates the flexibility and generalizability of our proposed method.

Table 11: The results of LLaMA-A and Bisecle with various LLM backbones.

Backbones	LLaMA-A		Bisecle	
	Acc	Fog	Acc	Fog
LLaMA-2-7B	46.58	13.83	62.37	5.34
Qwen-7B	60.77	6.54	63.97	3.57
Gemma-7B	58.44	8.48	61.26	5.88

B.6 Latency Comparison

We compared the training time of Bisecle and two variants (including the original model), which is shown in Table 12. From the results, we can see that the multi-directional supervision mechanism can lead to longer training time, while contrastive prompt learning only brings minor computational cost. It is reasonable because multi-directional supervision requires end-to-end model training on extra data, leading to increased computational overhead. Compared to the original model, the additional training time is acceptable, while the performance gain is substantial, demonstrating the efficiency of Bisecle.

Table 12: The training time required by Bisecle and two variants.

$\mathcal{L}_Q$	$\mathcal{L}_V$	$\mathcal{L}_P$	Time	Acc (↑)	Fog (↓)
✗	✗	✗	93 min	46.58	13.83
✓	✓	✗	171 min	59.78	6.58
✓	✓	✓	176 min	62.37	5.34

C Limitations and Broader Impacts

While Bisecle demonstrates significant potential in advancing continual learning for vision-language models through hippocampus-inspired mechanisms, our work primarily focuses on video understanding tasks. Future studies could extend this paradigm to other multimodal domains (e.g., audio-visual learning or embodied AI) and explore its scalability to diverse foundation models. Additionally, the current framework assumes task boundaries are known during training. Relaxing this assumption to enable true task-free continual learning remains an open challenge.

It should be noted that our method is not designed to outperform existing MLLMs, but rather to explore their potential in handling continually evolving data. This work provides preliminary insights for enabling LLMs personalization in dynamic environments. Moreover, given the promising results of binding and separation principles in mitigating forgetting, we believe this work opens new avenues for biologically-inspired learning systems in AI.