



Continual Text-to-Video Retrieval with Frame Fusion and Task-Aware Routing

Zecheng Zhao
The University of Queensland
Brisbane, Australia
uqzzha35@uq.edu.au

Zhi Chen
The University of Southern
Queensland
Toowoomba, Australia
zhi.chen@unisq.edu.au

Zi Huang
The University of Queensland
Brisbane, Australia
huang@itee.uq.edu.au

Shazia Sadiq
The University of Queensland
Brisbane, Australia
shazia@eecs.uq.edu.au

Tong Chen*
The University of Queensland
Brisbane, Australia
tong.chen@uq.edu.au

Abstract

Text-to-Video Retrieval (TVR) aims to retrieve relevant videos based on textual queries. However, as video content evolves continuously, adapting TVR systems to new data remains a critical yet under-explored challenge. In this paper, we introduce the first benchmark for Continual Text-to-Video Retrieval (CTVR) to address the limitations of existing approaches. Current Pre-Trained Model (PTM)-based TVR methods struggle with maintaining model plasticity when adapting to new tasks, while existing Continual Learning (CL) methods suffer from catastrophic forgetting, leading to semantic misalignment between historical queries and stored video features. To address these two challenges, we propose FrameFusionMoE, a novel CTVR framework that comprises two key components: (1) the Frame Fusion Adapter (FFA), which captures temporal video dynamics while preserving model plasticity, and (2) the Task-Aware Mixture-of-Experts (TAME), which ensures consistent semantic alignment between queries across tasks and the stored video features. Thus, FrameFusionMoE enables effective adaptation to new video content while preserving historical text-video relevance to mitigate catastrophic forgetting. We comprehensively evaluate FrameFusionMoE on two benchmark datasets under various task settings. Results demonstrate that FrameFusionMoE outperforms existing CL and TVR methods, achieving superior retrieval performance with minimal degradation on earlier tasks when handling continuous video streams. Our code is available at: <https://github.com/JasonCodeMaker/CTVR>.

CCS Concepts

• Information systems → Retrieval tasks and goals.

Keywords

Continual Text-to-Video Retrieval, Continual Learning, Video Representation Learning

*Corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License. *SIGIR '25, Padua, Italy*

© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1592-1/2025/07
<https://doi.org/10.1145/3726302.3729936>

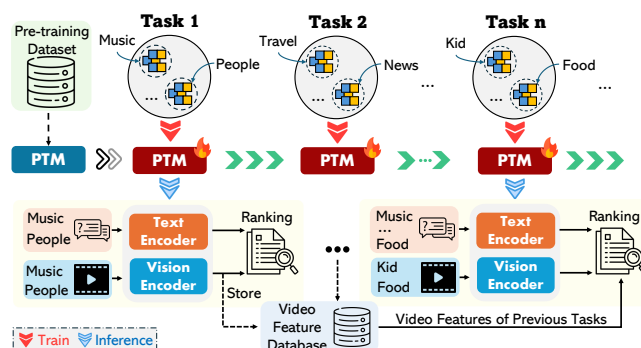


Figure 1: An illustration of Continual Text-to-Video Retrieval (CTVR) pipeline. A Pre-Trained Model (PTM) continuously adapts to a sequence of TVR tasks through continual learning. Video features extracted in the current task are stored in a database and leveraged for subsequent tasks. During inference, all task queries can retrieve relevant videos within the video feature database.

ACM Reference Format:

Zecheng Zhao, Zhi Chen, Zi Huang, Shazia Sadiq, and Tong Chen. 2025. Continual Text-to-Video Retrieval with Frame Fusion and Task-Aware Routing. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25)*, July 13–18, 2025, Padua, Italy. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3726302.3729936>

1 Introduction

The rapid growth of video-sharing platforms like YouTube has led to billions of Text-to-Video Retrieval (TVR) queries being processed daily [21]. With millions of new videos uploaded each day, these platforms continuously reflect evolving trends and shifting user interests. Meanwhile, the fast-paced content generation poses a unique challenge for TVR systems [13, 23, 28, 65, 76]: Continuous changes in data distribution make maintaining performance over time difficult for models. A naive solution of retraining TVR models with all the accumulated data is computationally expensive and difficult to scale. This raises a fundamental question: how can TVR systems adapt to new content over time without relying on historical data?

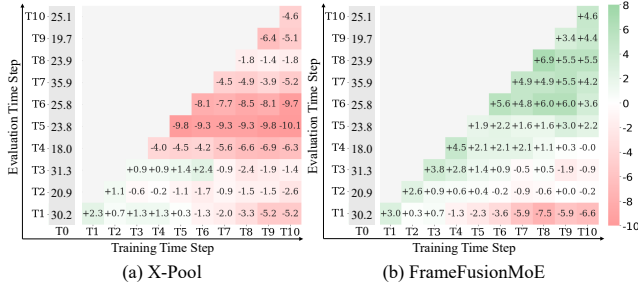


Figure 2: Visualization of model plasticity across sequential tasks (T), indexed chronologically. The first column T0 denotes the initial state of the pre-trained model without any updates. The presented results are performance variation on previous tasks after training on the current task (green for increase/red for drop) compared with the CLIP zero-shot results on MSRVT dataset. (a) The state-of-the-art TVR method X-Pool [18] exhibits declining plasticity to new tasks, i.e., underperform the zero-shot performance on the late stage tasks. (b) Our approach consistently improves task-wise performance while maintaining low backward forgetting when adapting to new tasks.

Continual Learning (CL) [22, 37, 48, 51] offers a promising solution for sequential tasks by enabling models to learn new tasks without forgetting previously acquired knowledge. Motivated by real-world challenges in dynamic video retrieval, we explore the application of CL to tackle the critical and underexplored problem of Continual Text-to-Video Retrieval (CTVR) as shown in Figure 1. In practical scenarios of video-sharing platforms, the dynamic interests will continuously drive new video categories (e.g., trending topics) along with corresponding text queries, where the text-video pairs form distinct tasks with varying distributions over time. Following standard practices in industry applications [8, 29], we resort to an offline computation strategy for generating video features for each task, while processing the text queries in real time. This approach is necessitated as (1) performing real-time inference for each video is computationally prohibitive, and (2) text queries, being highly dynamic and user-driven, can be efficiently processed on-the-fly using a text encoder. Under this setting, a CTVR model must learn to retrieve videos from these new categories using only current data, while ensuring it can retrieve videos across all previously learned categories during inference.

Despite advancements in CL [34, 36, 52, 54, 57] and TVR methods [18, 38, 68], CTVR poses unique challenges to these approaches. **(1) Model Plasticity Loss.** Existing TVR methods often fine-tune Pre-Trained Models (PTMs), such as CLIP [46], to improve video-text alignment. However, such extensive modifications to the joint visual-semantic embedding space compromise the model’s plasticity, reducing its ability to generalize to future tasks. For instance, X-Pool [18] introduces additional networks for learning joint video-text embeddings, while CLIP-ViP [68] proposes video proxies to account for temporal relationships. Taking X-Pool as an example in Figure 2(a), these methods require updating the entire model, which risks degrading transferability to future tasks as a result of overfitting to limited data early in training [39]. **(2) Catastrophic**

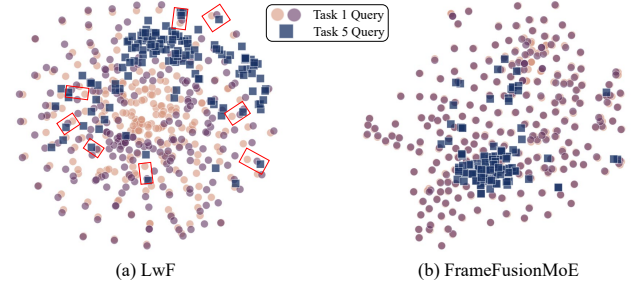


Figure 3: The catastrophic forgetting problem, illustrated by text feature shifts on MSRVT. Each $\bullet$ or $\blacksquare$ represents a query in Task 1 or Task 5, respectively. In addition, we use different colors to mark the states of Task 1 queries after each task update. Ideally, if there is no forgetting at all, each Task 1 query should have no movements in the embedding space after learning new tasks. (a) LwF [33], a strong CL baseline shows the query embeddings shift from the original position while model keeps updating, as highlighted by the scattered colors $\square$. (b) Our approach maintains stable features, with minimal shifts across tasks, as evidenced by the overlap among different colors.

Forgetting [16, 40, 41]: Most existing CL methods are designed for recognition tasks, where the goal is to classify inputs into pre-defined categories. These methods focus on making task-specific representations discriminative but largely ignore the need to maintain stable embedding correlations across tasks. Unlike recognition tasks, retrieval demands consistent alignment between text queries and video features over time. In CTVR, historical videos are stored as embeddings, requiring the text encoder to stay aligned with these embeddings when adapting to new tasks, such that text queries from earlier tasks can still be correctly mapped to relevant videos. However, as shown in Figure 3(a), current CL methods fail to address this requirement, i.e., the same text queries exhibit representation drift in the feature space, leading to overlapping query features across different tasks. This shows signs of catastrophic forgetting on the alignment between historical queries and stored video features, which degrades retrieval performance.

To address the challenges in CTVR, we propose FrameFusionMoE, a parameter-efficient continual learning framework that tackles both model plasticity loss and catastrophic forgetting. FrameFusionMoE comprises two core components: (1) the Frame Fusion Adapter (FFA) to capture temporal video dynamics and maintain frame-wise representations, and (2) the Task-Aware Mixture-of-Experts (TAME) to route textual queries to task-specific experts for mitigating query representation drift across tasks.

In FrameFusionMoE, FFA is designed to preserve the semantics of image-text feature space of CLIP, thereby maintaining the model’s plasticity and transferability to future tasks. Our preliminary experiments indicate that composing video features by simply averaging frame features retains generalization across tasks. However, this approach neglects the temporal relationships inherent in video frame sequences. To address this limitation, we propose to propagate frame features sequentially while ensuring that frame-wise features remain intact. Specifically, we integrate the FFA into

each Transformer block of the CLIP image encoder, where each FFA is implemented with a multi-head cross attention unit that takes previous frame features as the query and current frame features as the key and the value. As reflected by Figure 2(b), this enables the model to effectively capture temporal dependencies without compromising the generalizability of the joint image-text feature space of CLIP.

To tackle the catastrophic forgetting on the misalignment between historical queries and the stored video features, we introduce TAME. It preserves previously learned representations while allowing the model to adapt to new tasks by selectively routing text queries to the appropriate experts. The architecture is built on a Mixture-of-Experts (MoE) design, where each expert is implemented using LoRA-like layers [26, 53]. To achieve task-specific routing, we propose a task-aware router that constructs task prototypes, which guide the model in selecting the most relevant expert for handling task-dependent queries. This design minimizes representation drift by ensuring that embeddings from previous tasks remain aligned with their corresponding queries across time.

Overall, the contributions of this paper are:

- We propose a FrameFusionMoE framework for Continual Text-to-Video Retrieval (CTVR), a practical yet under-explored area of video understanding. **To the best of our knowledge, FrameFusionMoE is the first attempt for CTVR.**
- We design two novel components in FrameFusionMoE to address the unique challenges in CTVR: Frame-Fusion Adapter (FFA) and Task-Aware Mixture of Experts (TAME). The FFA preserves the image-text embedding space of Pre-Trained Models (PTMs) while capturing temporal video dynamics, ensuring model plasticity for future tasks. TAME mitigates catastrophic forgetting by using a task-aware routing mechanism to maintain consistent alignment between text queries and stored video features, ensuring long-term retrieval performance.
- We benchmark CTVR on two text-to-video datasets, *i.e.*, MSRVTT [67] and ActivityNet [2]. We re-purpose and evaluate four state-of-the-art continual learning (CL) and three text-to-video retrieval (TVR) baselines for CTVR. Our extensive benchmarking results highlight the limitations of existing methods and verify the advantageous performance of FrameFusionMoE.

2 Related Work

Text-to-Video Retrieval Deep learning has revolutionized computer vision [4–7, 24, 56], creating a foundation for advances in Text-to-Video Retrieval (TVR) [12, 13, 19, 28, 32, 65, 70, 76] through pre-trained Vision-Language Models like CLIP [46], which bridge textual descriptions and visual content. CLIP4Clip [38] pioneered the application of CLIP models for text-to-video retrieval tasks, demonstrating CLIP’s robust transfer learning capabilities. However, recognizing that videos contain unique temporal information that static images lack, there is a significant domain gap between videos and images. Many methods [9, 14, 18, 35, 38, 59, 66, 68] have been proposed to leverage temporal dynamics to enhance video representations. X-Pool [18] leverages text-conditioned feature attention across video frames to generate semantically enriched embeddings. Meanwhile, TS2-Net [35], CLIP-ViP [68], and Prompt Switch [9] enhance video representations by incorporating temporal or video-specific embeddings to capture inter-frame

relationships. Unlike video representation learning, T-MASS [59] utilizes stochastic text embeddings to strengthen video-text alignment. Although these architectural modifications to CLIP enable effective adaptation to video-language alignment, they compromise CLIP’s original generalization capabilities. Our proposed Frame Fusion Adapter (FFA) enhances temporal dynamics of video frames while preserving model plasticity for new tasks.

Continual Learning. Continual Learning (CL) [20, 22, 37, 45, 48, 51, 73] is a machine learning paradigm where models learn sequentially from a stream of tasks while maintaining performance on previous tasks. Class Incremental Learning (CIL) [25, 47, 55] represents a challenging paradigm where each task contains non-overlapping categories, and test samples may come from any previous task, requiring models to balance stability and plasticity. Recently, some task-specific benchmarks [17, 34, 36, 49, 52, 54, 57] have been developed that enable standardized evaluation protocols between different CL methods. In image domain, Core50 [36], CLeAR [34] and CLiMB [52] compare and analyze various CL methods from different perspectives. Extending to the video domain, vCLiMB [57] introduces the first benchmark for continual learning in video action recognition, while ViLCo-Bench [54] subsequently focused on evaluating continual learning tasks in the video-language domain. However, Text-to-Video Retrieval (TVR) remains underexplored despite its practical importance. In this work, we benchmark continual learning for text-to-video retrieval.

Continual Learning with Pre-Trained Models. In traditional CL approaches, models learn incrementally from sequential tasks, often leading to overfitting on the initial task. Fortunately, with the advent of large-scale pre-trained models (PTMs), the field has shifted its focus towards leveraging their robust representation capabilities [42, 75, 78]. These emerging approaches can be categorized into three main paradigms: prompt-based methods [30, 44, 60, 62, 63, 69], regularization-based methods [10, 33, 77], and model mixture-based methods [3, 61, 64, 71, 79]. The prompt-based methods leverage the strong generalization capabilities of PTMs by introducing minimal trainable prompt parameters, enabling efficient adaptation. L2P [63] maintains a prompt pool and selects the most relevant prompts for each test sample. DualPrompt [62] incorporates additional task-specific prompts, enabling the model to encode both task-invariant patterns and task-specific instructions. Moreover, PIVOT [58] integrates prompting mechanisms into video CL for adaptive prompt selection. Regularization-based methods introduce regularization terms to achieve a balance between stability and plasticity. EWC [31], SI [74] and MAS [1] employ parameter-specific regularization terms that add a penalty to the weight updates based on each parameter’s importance for previously learned tasks. On the other hand, LwF [33] mitigates catastrophic forgetting with knowledge distillation by treating the previous model as a teacher and the current model as a student. LwF-VR [10] utilizes the CLIP vocabulary set as a reference, while ZSCL [77] leverages ImageNet as a reference dataset, both aiming to better preserve the pre-trained model’s capabilities. In contrast to training-phase optimization, model mixture-based methods tackle catastrophic forgetting during inference by combining experts from different tasks. MoE-Adapter [71] incorporates Mixture-of-Experts (MoE) [15, 27, 50] as specialized adapters. Each expert is trained to handle distinct knowledge

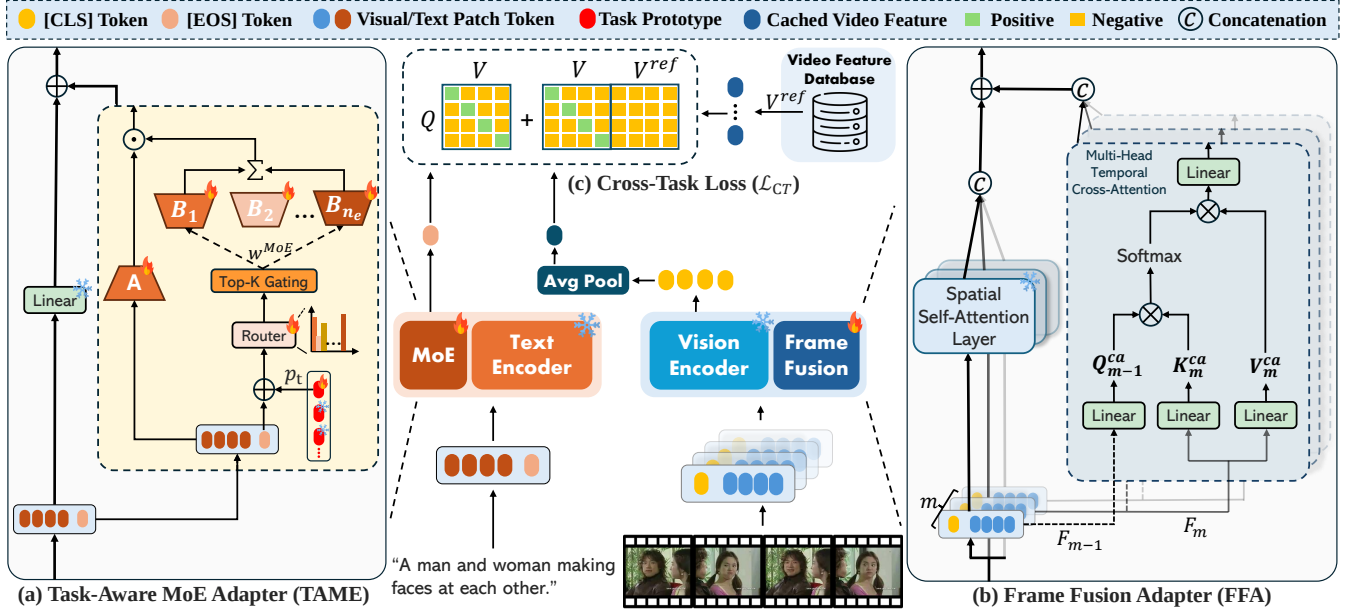


Figure 4: Overall framework of the FrameFusionMoE. It consists of three core components: (a) A Task-Aware MoE Adapter (TAME) that is added to a frozen CLIP text encoder to learn the distribution of text query through the selection of multiple experts $\{B_i\}_{i=1}^{n_e}$. The expert weights w^{MoE} are determined by a router taking the element-wise addition ($\oplus$) of the [EOS] token and task prototype p_t as input. (b) A vision processing pipeline where frame features are processed through a frozen CLIP vision encoder and Frame Fusion Adapters (FFA). Each FFA uses previous frame feature maps F_{m-1} to attend over current frame F_m through multi-head temporal cross attention. The FFA output serves as a temporal guidance signal that is added back to each spatial self-attention layer. (c) The Cross-Task Loss ($\mathcal{L}_{CT}$) optimizes representations by drawing matched text-video pairs closer while pushing away cached video features that serve as negative samples.

distributions. However, TVR poses a unique catastrophic forgetting challenge on the alignment between historical queries and stored video features. To cope with such unique challenge, our proposed task-aware mixture-of-experts can maintain the historic queries distribution while adapting to new tasks.

3 Continual Text-to-Video Retrieval

In this section, we first formalize the practical yet under-studied research problem of learning a text-to-video retrieval system in sequential tasks, *i.e.*, Continual Text-to-Video Retrieval (CTVR). Then, we discuss the motivation of the proposed method. Lastly, we introduce the components of our proposed FrameFusionMoE.

3.1 Problem Definition

In CTVR, we aim to train a retrieval model that allows text queries q to retrieve relevant videos v while videos of new tasks emerge. Given a sequence of T tasks $\{D_1, \dots, D_T\}$, where each task $D_t = \{(q_i^t, v_i^t)\}_{i=1}^{n_t}$ contains n_t query-video pairs from categories C_t , where $C_i \cap C_j = \emptyset, \forall i \neq j$. During training on task t , the data access is restricted to only D_t , while the data of all previous tasks $\{D_1, \dots, D_{t-1}\}$ are inaccessible. For each query-video pair, we uniformly sample M frames from a video $v = [f_1, f_2, \dots, f_M]$, each is extracted as frame features $v = [f_1, f_2, \dots, f_M]$ by the CLIP image encoder. f_m represents the [CLS] token from the Transformer-based vision-encoder for m -th frame. The text encoder takes the [EOS] token as query features q . After training on a task t , the testing video

features $V_t = \{v_i\}_{i=1}^{n_t}$ of categories C_t are saved into the database $V_{[1:t]} = V_{[1:t-1]} \cup V_t$. During testing at task t , given a set of test queries $Q = \bigcup_{i=1}^t Q_i$ from all previous tasks, the model retrieves relevant videos from the database V . For each query $q \in Q$, the retrieval is performed by computing the cosine similarity $\text{sim}(q, v)$ between the query features q and the video features v . The videos are then ranked according to $\text{rank}(v) = \text{sort}_{v \in V_{[1:t]}}(\text{sim}(q, v))$ in descending order.

3.2 Motivation

To explore the key aspects of an effective CTVR system, we analyze state-of-the-art methods for TVR and CL, so as to locate the research challenges when two areas intersect.

Recent TVR methods primarily adapt the knowledge of Pre-Trained Models (PTMs) from the image-text domain to the video-text domain. These adaptations often require significant modifications to the joint image-text embedding space to account for video temporality. For instance, X-Pool [18] introduces additional networks for learning joint video-text embeddings. CLIP-ViP [68] incorporates video proxy tokens to account for temporal relationships. However, as shown in Table 1, the TVR baselines underperform the Average Pooling baseline [38] which simply averages frame features to represent video features. We hypothesize that the adaptations are designed as one-off solutions, focusing solely on the current task while failing to retain the pretraining knowledge of PTMs. This results in **model plasticity loss**, hindering the model's ability to adapt to future tasks.

In CTVR, re-extracting video features from historical data after each task is computationally prohibitive. A feasible alternative is to cache historical video features into a database after each task. Thus, an effective CTVR system must maintain semantic alignment between historical queries and these cached video features while learning new tasks. Unfortunately, existing CL methods are predominantly designed for recognition tasks, where the focus is on classifying inputs into predefined categories. These methods prioritize task-specific discriminative features but neglect the need to maintain stable text query features across tasks. For example, Mixture-of-Experts Adapter (MoE-Adapter) [71] has demonstrated continual adaptation to sequential tasks, where the experts are dynamically updated for new tasks. This leads to **catastrophic forgetting**, where the alignment between historical queries and stored video features deteriorates.

Our framework consists of: (1) the Frame Fusion Adapter (FFA) that captures video temporality while preserving CLIP’s image-text embedding space, ensuring model plasticity for future tasks; and (2) the Task-Aware Mixture-of-Experts (TAME) that maintains alignment between historical queries and cached video features, mitigating catastrophic forgetting in the embedding space. In what follows, we unfold the design of those core components.

3.3 Frame Fusion Adapter

The Frame Fusion Adapter (FFA) is designed to preserve CLIP’s model plasticity while enabling the learning of video temporality in a parameter-efficient manner. Following the design principles of AvgPool [38], Frame Fusion maintains the joint image-text embedding space of CLIP without introducing disruptive modifications, thereby retaining the model’s pre-trained generalization capabilities. To achieve this, FFAs are introduced as lightweight adapters that are placed between the Transformer blocks of CLIP, while keeping the PTM parameters frozen. This approach minimizes computational overhead while allowing the model to adapt to video-specific tasks. Furthermore, FFAs enable dependent frame features to propagate across frames, effectively capturing the temporal dynamics inherent in videos.

As shown in Figure 4(b), FFAs are implemented with Cross-Attention (CA) blocks in parallel with existing Self-Attention (SA) blocks in CLIP. Essentially, the input image tokens of SA are simultaneously fed into both SA and CA blocks, and the output from both are added together for the following Transformer blocks. Specifically, consider the image tokens of each frame $[F_1, F_2, \dots, F_M]$, where M is the number of frames. Then, for the m -th frame in a video, CA can be formulated as:

$$Q_{m-1}^{ca} = F_{m-1}W_q^T, \quad K_m^{ca} = F_mW_k^T, \quad V_m^{ca} = F_mW_v^T, \quad (1)$$

$$A^{ca} = \text{softmax}(Q_{m-1}^{ca}K_m^{caT}/\sqrt{O/h}), \quad CA(F_{m-1}, F_m) = A^{ca}V_m^{ca},$$

where the attention query comes from the patch tokens of the previous frame F_{m-1} and the key/values are the current frame F_m . W_q, W_k and $W_v \in \mathbb{R}^{O \times (O/h)}$ are trainable linear layers, O and h are the feature dimension and number of heads. The CA block is implemented in parallel to the existing SA blocks in CLIP. Similarly, SA blocks can be formulated as:

$$Q_m^{sa} = F_m\tilde{W}_q^T, \quad K_m^{sa} = F_m\tilde{W}_k^T, \quad V_m^{sa} = F_m\tilde{W}_v^T, \quad (2)$$

$$A^{sa} = \text{softmax}(Q_m^{sa}K_m^{saT}/\sqrt{O/h}), \quad SA(F_m, F_m) = A^{sa}V_m^{sa},$$

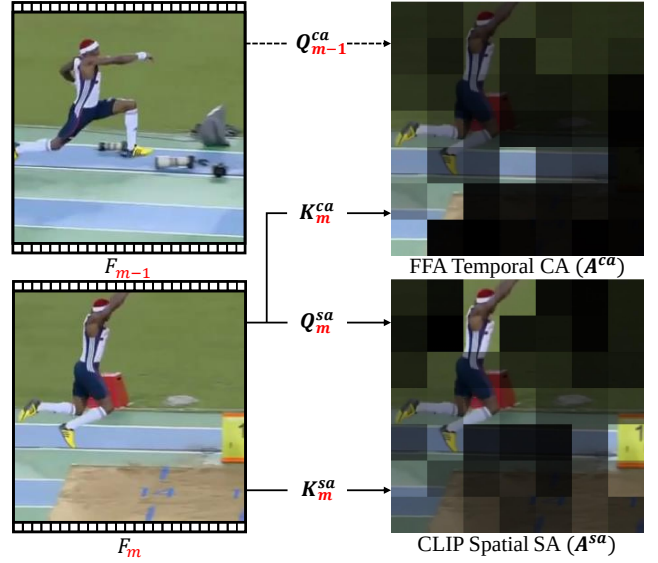


Figure 5: Visualization of attention maps from FFA Temporal CA and CLIP Spatial SA mechanisms. Brighter regions in the attention maps indicate higher attention weights. FFA’s temporal CA demonstrates stronger attention weights on temporally consistent regions between frames (e.g., track surface, background) while showing lower attention on the changing sand pit area, effectively capturing inter-frame consistency. CLIP’s spatial SA focuses on the athlete and their jumping action, capturing semantically important motion information within the frame.

The fused tokens are added with the original tokens via residual connections:

$$F_{m*} = SA(F_m, F_m) + \alpha CA(F_{m-1}, F_m) \quad (3)$$

where α is a trainable parameter that controls the weight of CA. The obtained new frame tokens F_{m*} is further fed into the following Transformer blocks and after each, we have an FFA module applied.

The SA blocks primarily capture intra-frame patch relationships, while the CA blocks focus on inter-frame temporal dynamics by integrating features from adjacent frames. By combining these mechanisms, FFAs efficiently utilize frame-level features extracted at each CLIP layer to enrich video representations with temporal information. To gain further insights into the behavior of FFAs, we visualize the attention matrix A^{ca} and A^{sa} . As shown in Figure 5, the CA attention query from the previous frame Q_{m-1}^{ca} , predominantly attends to temporally consistent regions across frames such as the track surface and background elements. This behavior ensures inter-frame consistency while de-emphasizing regions undergoing rapid changes, such as athlete’s movement. In contrast, the SA mechanism primarily focuses on the dynamic foreground elements, effectively capturing semantically important motion information within the frame. This combination of intra-frame and inter-frame attention mechanisms facilitates effective video representation learning through Frame Fusion while preserving the original architecture of CLIP. Consequently, the proposed approach ensures both computational efficiency and strong generalization across continual learning tasks.

3.4 Task-Aware Mixture-of-Experts Adapter

To mitigate the catastrophic forgetting problem caused by misalignment between historical queries and stored video features, we introduce Task-Aware Mixture-of-Experts (TAME) adapters. The primary objective of TAME is to learn task-conditional text features, ensuring alignment across sequential tasks while maintaining efficient adaptation.

As illustrated in Figure 4(a), TAME adapters are integrated with the linear layers in the SA blocks of the CLIP text encoder. A TAME adapter consists of a set of expert networks $\{E_i(\cdot)\}_{i=1}^{n_e}$ where n_e represents the number of experts, a router function $R(\cdot)$, and task prototypes $\{p_t\}_{t=1}^T$.

Given an input textual query with n_x tokens $X = [x_1, x_2, \dots, x_{n_x}, x^{EOS}]$, where x^{EOS} is the end-of-sentence token representing the text features of the entire sentence. During task t , the token x^{EOS} is added with the current task prototype p_t . The resulting features are passed into the router to produce the gating parameter w^{MoE} , which determines the activation of experts:

$$w^{MoE} = \text{softmax}(\text{TopK}(R(x^{EOS} + p_t))), \quad (4)$$

where TopK are the experts with highest confidence. The selected K experts will take the tokens X as input. Specifically, the experts are parameterized using a LoRA structure, where the low-rank encoder A is shared among experts while the expert-specific decoders are represented as $\{B_i\}_{i=1}^{n_e}$. The experts output is then formulated as:

$$E(X) = \sum_{i \in \text{TopK}} w^{MoE} \cdot (B_i A X). \quad (5)$$

Let $\tilde{W}$ be the frozen layers in the SA blocks of CLIP text encoder, the output of the layer coupled with TAME is formulated as:

$$X^* = X\tilde{W}^T + \lambda E(X), \quad (6)$$

where λ controls the scale of the residual features.

During inference, for a given textual query, each task prototype p_i (where $i \in \{1, \dots, T\}$) is added to the end-of-token x^{EOS} , resulting in task-conditional text features $\{q_i\}_{i=1}^T$. These features are then used to compute similarity scores with the stored video features from different tasks $V_{[1,T]}$, and the TopK most relevant videos are retrieved.

3.5 Optimization

To effectively align representations between video and textual modalities based on a pre-trained CLIP model, we optimize cross-modal alignment in a shared embedding space for each task. They are video-to-text ($v2t$) and text-to-video ($t2v$) modalities:

$$\begin{aligned} \mathcal{L}_{v2t} &= -\mathbb{E}_i \log \frac{\exp(\langle q_i, v_i \rangle / \tau)}{\sum_j \exp(\langle q_j, v_i \rangle / \tau)}, \\ \mathcal{L}_{t2v} &= -\mathbb{E}_i \log \frac{\exp(\langle q_i, v_i \rangle / \tau)}{\sum_j \exp(\langle q_i, v_j \rangle / \tau)}. \end{aligned} \quad (7)$$

We also introduce a Cross-Task (CT) loss that leverages the samples within the video feature database as negative references. By considering videos from previous tasks, this loss performs semantic regularization on the alignment between queries and their relevant videos. It maintains task-specific feature distributions while preventing catastrophic forgetting. We formulate CT loss as:

$$\mathcal{L}_{CT} = -\mathbb{E}_i \left[\log \frac{\exp(\langle q_i, v_i \rangle / \tau)}{\sum_t \exp(\langle q_i, v_t \rangle / \tau) + \sum_h \exp(\langle q_i, v_h^{ref} \rangle / \tau)} \right] \quad (8)$$

where q_i and v_i denote the text query and its matched video features from the current task, v_t represents current task video features as in-batch negatives, v_h^{ref} denotes video features of previous tasks serving as additional negative references, and τ is a temperature parameter. The overall objective for cross-modal alignment between text and video representations is defined as:

$$\mathcal{L} = (1 - \beta) \frac{1}{2} (\mathcal{L}_{v2t} + \mathcal{L}_{t2v}) + \beta \mathcal{L}_{CT}, \quad (9)$$

where β is a pre-defined hyper-parameter with $\beta = 0$ for task 1.

4 Benchmark Experimental Setup

We first present the experimental setup for CTVR benchmark, including datasets (Section 4.1), evaluation metrics (Section 4.2), baselines (Section 4.3), and implementation details (Section 4.4). Then, we conduct comprehensive experiments to address each research question in Section 5.

4.1 Datasets

We construct CTVR using two established TVR datasets with pre-defined categorical structures. To comprehensively assess CL capabilities, we evaluate models across two settings with 10 and 20 sequential tasks. (1) **MSRVTT** [67] consists of 10,000 videos (10-32 seconds) and 200,000 captions across 20 distinct classes. Traditional TVR works utilize *Train-7K* [43] for training and 1K-A test set [72] for evaluation. (2) **ActivityNet Captions** [2] consists of 20,000 video clips (average 180 seconds) and 100,000 descriptions across 200 categories. Different from traditional TVR evaluation pipelines that concatenate all descriptions for paragraph-video retrieval, we utilize the trimmed subset [2] and employ LLMs [11] to select the most category-representative description-video clip pair from each video. This single-pair selection better aligns with real-world search scenarios while maintaining retrieval complexity.

4.2 Evaluation Metrics

Following standard TVR evaluation [18, 38], we measure how well the model performs in retrieval across all learned tasks by reporting Recall@1 (**R@1**), Recall@5 (**R@5**), Recall@10 (**R@10**), Median Rank (**MedR**) and Mean Rank (**MeanR**). When evaluating at task t , we test the model on queries from both current and all previous tasks $Q_{[1:t]}$. The search space consists of all videos from the video feature database $V_{[1:t]}$, where $V_{[1:t-1]}$ are extracted and stored using the models learned in previous tasks. Videos are ranked by cosine similarity between query and video features.

To measure how learning new tasks affects the model's performance on previous tasks, we evaluate **Backward Forgetting (BWF)**. When testing on task t , we measure the performance drop of each previous task i (where $i < t$) by comparing its current performance with its performance right after the task was initially learned. Formally, the BWF at task n is defined as $\text{BWF}_t = \frac{1}{t-1} \sum_{i=1}^{t-1} (R_{i,i} - R_{t,i})$, where $R_{i,i}$ represents the R@1 on the queries of task i after training on task i , and $R_{t,i}$ measures the R@1 on the queries of task i after training on task t .

Table 1: Comparison of model performance for CTVR on MSRVT and ACTNET datasets with 10 and 20 tasks, respectively. The top two models are highlighted in bold and underlined. ‘E.R.’ means the utilization of an external reference dataset.

Model	E.R.	MSRVT-10							MSRVT-20							ACTNET-10							ACTNET-20							Params		
		Recall			Rank			BWF ↓	Recall			Rank			BWF ↓	Recall			Rank			BWF ↓	Recall			Rank			BWF ↓			
		@1 ↑	@5 ↑	@10 ↑	Med ↓	Mean ↓	Mean ↓		@1 ↑	@5 ↑	@10 ↑	Med ↓	Mean ↓	Mean ↓		@1 ↑	@5 ↑	@10 ↑	Med ↓	Mean ↓	Mean ↓		@1 ↑	@5 ↑	@10 ↑	Med ↓	Mean ↓	Mean ↓				
Zero-Shot CLIP [46]	×	22.14	41.24	51.34	10.00	117.48	0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	14.89	34.97	47.78	12.00	84.02	0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	0.00M			
CLIP4Clip [38]	×	23.57	44.76	54.48	8.00	80.23	0.61	±0.37	±0.24	±0.41	±0.00	±1.32	0.37	±0.20	±0.30	±0.35	±0.00	±0.47	±0.46	±0.86	±0.88	±0.33	±0.58	±0.92	±0.09	±0.57	±0.45	±0.16	±0.41	±0.22	151.28M	
X-Pool [18]	×	19.60	39.80	49.49	11.00	94.89	0.28	±0.35	±0.63	±0.54	±0.00	±1.70	0.40	±0.75	±1.52	±0.87	±0.58	±1.97	±0.23	±0.54	±0.87	±1.06	±0.58	±0.39	±0.38	±0.47	±0.35	±0.19	±1.58	±0.43	152.59M	
CLIP-ViP [68]	×	21.56	44.19	53.43	8.00	86.71	0.49	±1.07	±0.31	±0.52	±0.00	±0.71	±0.74	±0.19	±0.29	±0.20	±0.00	±1.19	±0.31	±0.53	±0.82	±0.75	±0.58	±1.50	±0.21	±0.19	±0.21	±0.71	±0.38	±1.16	±0.13	151.29M
LwF [33]	✓	23.85	45.30	55.68	7.33	76.46	1.68	±0.09	±0.26	±0.32	±0.58	±0.44	±0.59	±0.44	±1.33	±0.91	±1.00	±3.99	±0.72	±0.12	±0.20	±0.41	±0.00	±1.86	±0.42	±0.31	±0.31	±0.46	±0.25	±2.22	±0.14	151.28M
VR-LwF [10]	✓	24.49	45.59	55.45	7.33	74.89	1.22	±0.20	±0.14	±0.89	±0.58	±2.56	±0.46	±0.43	±0.43	±0.96	±0.58	±2.16	±0.16	±0.55	±0.36	±0.45	±0.50	±2.39	±0.47	±0.36	±0.26	±0.15	±0.13	±0.70	±0.13	151.28M
ZSCL [77]	✓	23.99	45.15	54.77	8.00	79.69	0.10	±0.44	±0.33	±0.24	±0.00	±0.95	±0.78	±0.47	±0.61	±0.25	0.93	±0.88	±0.91	±0.67	±0.41	±0.05	±0.00	±0.19	±0.35	±0.63	±0.89	±0.07	±0.33	±65.03	±0.70	151.28M
MoE-Adapter [71]	×	22.92	42.76	52.11	9.00	105.70	0.14	±0.09	±0.24	±0.09	±0.00	±2.66	±0.11	±0.14	±0.16	±0.10	±0.00	±0.34	±0.05	±0.55	±0.48	±0.89	±0.58	±3.70	±0.08	±0.11	±0.14	±0.24	±0.00	±1.23	±0.27	59.8M
TVR [68] + CL [10]	✓	22.47	43.71	53.59	8.00	82.97	0.46	±0.55	±0.42	±0.68	±0.00	±1.31	±0.17	±0.66	±0.11	±0.63	±0.58	±3.44	±0.73	±0.62	±0.70	±1.45	±0.58	±4.34	±0.27	±0.27	±1.08	±1.11	±0.58	±5.46	±0.26	151.29M
StableFusion	×	25.87	45.91	56.03	7.00	74.70	-0.45	±0.33	±0.03	±0.64	±0.00	±1.06	±0.22	±0.14	±0.42	±0.07	±0.58	±0.57	±0.25	±0.33	±0.29	±0.34	±0.00	±0.63	±0.40	±0.25	±0.08	±0.15	±0.00	±1.99	±0.06	46.8M
Upper Bound	×	25.86	48.34	58.96	6.00	65.10	-0.17	±0.67	±0.25	±0.12	±0.00	±1.09	±1.21	±0.48	±0.54	±0.65	±0.00	±1.79	±0.44	±0.88	±0.17	±0.23	±0.00	±0.18	±0.10	±0.17	±0.31	±0.24	±0.00	±0.76	±0.31	151.28M

4.3 Benchmarking Baselines

4.3.1 TVR Methods. Building upon CLIP’s [46] vision-language embedding capabilities, recent TVR models aim to transfer knowledge from the image-text domain to the video-text domain.

Parameter-free Methods. CLIP4Clip [38] introduces the first parameter-free TVR model (AvgPool) based on CLIP, where video features are obtained by decomposing videos into image sequences and applying average pooling on a sequence of frame features. **Upper Bound** represents the performance ceiling based on CLIP4Clip [38], where the model is trained with full access to the combined data from all tasks simultaneously in an i.i.d. manner.

Architecture-enhanced Methods. These approaches focus on architectural modifications on CLIP to enable video temporal feature learning capabilities, thereby enhancing video-text representations. **X-Pool** [18] introduce a Transformer block on top of the vision-language embedding space that learns the relevance between queries and individual video frames. **CLIP-ViP** [68] introduces a lightweight architectural modification that augments CLIP’s temporal learning capacity through the incorporation of temporal embeddings and video proxy vectors.

4.3.2 Vision-Language CL Methods. Our benchmark includes CL approaches that are established with CLIP.

Knowledge Distillation-based Methods. LwF [33] utilizes the model from the previous task as a teacher model, employing distillation loss to maintain feature preservation for older tasks during current task learning. **VR-LwF** [10] extends the LwF framework with a focus on maintaining CLIP’s inherent text-vision alignment capabilities. Through random sampling from CLIP’s vocabulary set to create a replayed vocabulary set during training, it achieves better mitigation of catastrophic forgetting. **ZSCL** [77] employs a reference dataset to preserve CLIP’s original image-text alignment.

Dynamic Network Methods. MoE-Adapter [71] introduces a MoE-structured adapters to preserve the zero-shot capabilities of vision-language models. Through routing and gating operations, MoE dynamically selects appropriate experts.

To eliminate confounding factors from TVR model architectures, our benchmark implements these CL baselines on the **CLIP4Clip**,

maintaining maximum consistency with their original deployment methodologies. In addition, we also consider the combination between **VR-LwF** and **CLIP-ViP**.

4.4 Implementation Details

For all experiments, we utilize CLIP-B/32 as the pre-trained model with 20 training epochs per task and 16 videos per class. All methods employ a cosine learning rate scheduler. For baseline-specific parameters, CLIP-ViP [68] is configured with 4 video proxies and methods involving knowledge distillation [10, 33, 77] are implemented with a temperature of $\tau = 2.0$. All methods are evaluated with mean and standard deviation across three different random seeds. For experiments on MSRVT and ACTNET datasets, we sample 12 and 24 frames per video with batch sizes of 8 and 16, respectively. For our method, we employ learning rates of 4×10^{-6} and 6×10^{-6} , respectively. The frame fusion adapter consists of 10 layers for MSRVT and 12 layers for ACTNET, with each TAME layer containing 10 experts for MSRVT and 5 experts for ACTNET, and a loss scale of $\beta = 0.6$.

5 Results and Analysis

To comprehensively evaluate the effectiveness of our proposed method, we conduct extensive experiments to answer the following research questions: (1) **RQ1:** How does FrameFusionMoE perform compared to existing TVR and CL baselines in terms of effectiveness on CTVR? (2) **RQ2:** How do the main architectural components of FrameFusionMoE contribute to its overall performance? (3) **RQ3:** How do key hyper-parameters of FrameFusionMoE affect its continual retrieval performance?

5.1 Overall Performance (RQ1)

Table 1 and Figure 6 present a comparative analysis of the CTVR performance and backward forgetting between FrameFusionMoE and the baseline approaches across two datasets with four CL configurations. We observe the following key observations:

Effectiveness of FrameFusionMoE. Across all CTVR configurations, FrameFusionMoE surpasses all baseline models, showing

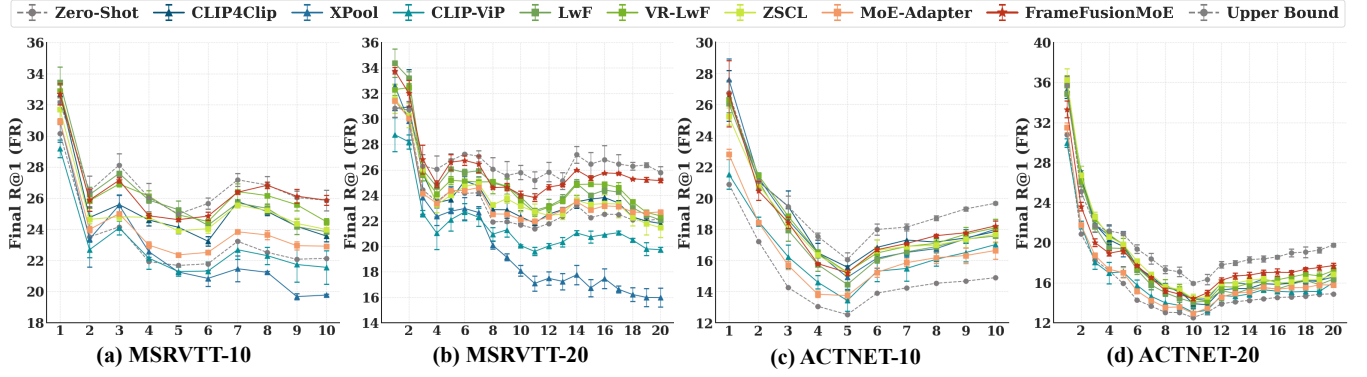


Figure 6: Comparative analysis of Final R@1 (FR) performance for different CTVR configurations across all tasks.

R@1 improvements of +1.38, +2.46, +0.13, and +0.50 over the second-best model on all datasets. Figure 6 demonstrates that our model achieves higher R@1 than all baselines across different CTVR configurations and tasks. Notably, our method tends to establish higher performance gain when learning through more tasks, which demonstrates the continual learning capability in long-term tasks.

Our method achieves near-zero BWF scores (-0.70 to $+0.04$) across multiple CL configurations, indicating minimal catastrophic forgetting. Notably, in several configurations, we observe negative BWF scores, suggesting that learning new tasks enhances the model’s performance on previous tasks. FrameFusionMoE achieves these improvements while maintaining computational efficiency. Compared to standard CL baselines, FrameFusionMoE involves only 30.94% of the trainable parameters. Compared to the parameter-efficient MoE-Adapter [71], we use 70.26% of its parameters.

Model Plasticity Loss of TVR Methods. TVR baselines exhibit unexpected performance patterns in CTVR benchmark compared to the one-off learning scenario. Our experiments show that the CLIP4clip, which preserves CLIP’s original architecture, consistently outperforms more advanced TVR models across various CL configurations. This counter-intuitive result is particularly pronounced in MSRVTT. As shown in Figure 2, carefully designed architectures like XPool and CLIP-ViP significantly degrade CLIP’s original generalization capability after learning the first few tasks, making it difficult to adapt to subsequent tasks with limited training data. This loss of model plasticity cascades through the continual learning process.

Catastrophic Forgetting on CL Baselines. Our experimental results indicate that these Continual Learning (CL) baselines demonstrate less significant improvements in TVR tasks compared to their established performance gains in image recognition [10, 33, 71, 77]. This performance disparity stems from fundamental differences in objectives: retrieval tasks are particularly susceptible to Representation Shift, which significantly impacts performance in CTVR scenarios. As demonstrated in Figure 3, fine-tuning on new tasks causes text embedding representation shift, leading to a progressive misalignment between video and query representations in the shared embedding space. This explains why CL baselines such as LwF experience a performance drop of 2.1 as the number of tasks increases from 10 to 20 on the MSRVTT.

Table 2: Ablation study of the contribution of individual components in our proposed method. We report the performance impact of removing Frame Fusion Adapter (FFA), Task-Aware Mixture-of-Experts (TAME), Task-Specific Prototype (TP) modules and Cross-Task Loss ($\mathcal{L}_{CT}$). The Δ represent performance degradation relative to the complete model highlighted in red.

Model	MSRVTT						ActivityNet					
	10 Task			20 Task			10 Task			20 Task		
	R@1 $\uparrow$	Δ	BWF $\downarrow$	R@1 $\uparrow$	Δ	BWF $\downarrow$	R@1 $\uparrow$	Δ	BWF $\downarrow$	R@1 $\uparrow$	Δ	BWF $\downarrow$
Ours	26.25	-0.69	25.32	-0.86	18.49	-0.26	17.92	0.08				
w/o FFA	17.59	-8.66	0.13	20.60	-4.72	-0.21	16.84	-1.65	-0.28	16.42	-1.50	-0.44
w/o TAME	25.22	-1.03	-0.04	24.58	-0.74	0.62	16.13	-2.36	0.27	15.76	-2.16	-0.43
w/o TP	25.92	-0.33	-0.98	24.78	-0.54	-0.80	17.72	-0.77	0.39	17.90	-0.02	-0.29
w/o $\mathcal{L}_{CT}$	25.15	-1.10	-0.09	24.21	-1.11	-0.16	17.70	-0.79	0.42	17.27	-0.65	0.38
Baseline [38]	23.18	-3.07	0.96	21.87	-3.45	1.15	17.92	-0.57	0.76	17.11	-0.81	0.46

5.2 Effectiveness of Components (RQ2).

To thoroughly validate the effectiveness of our proposed method, we conduct an ablation study on the key components of our architecture. In Table 2, we respectively remove individual components to measure their contribution to the entire framework. **FrameFusion Adapter (FFA):** The FFA module enables efficient temporal video learning, and its removal causes model plasticity loss. **Task-Aware Mixture-of-Experts (TAME):** The TAME module maintains query-feature alignment across tasks, and its removal leads to catastrophic forgetting in the shared feature space. **Task-Prototype (TP):** The TP module learns task-specific features to guide TAME’s routing, and its removal impairs the model’s task discrimination capability. **Cross-Task Loss ($\mathcal{L}_{CT}$)** maximizes the distance between current query-video features and cached video representations in the feature space, and its removal causes feature overlap across different tasks.

Effectiveness of Individual Components. Our ablation studies demonstrate that every component contributes to model performance, as removing any component results in significant performance degradation across different tasks. The ablation study demonstrates the crucial role of each proposed component. First, removing FFA causes the most severe performance drop, underscoring its fundamental importance in maintaining model plasticity. Second, the absence of TAME leads to increased backward forgetting, validating its effectiveness in mitigating catastrophic forgetting. The

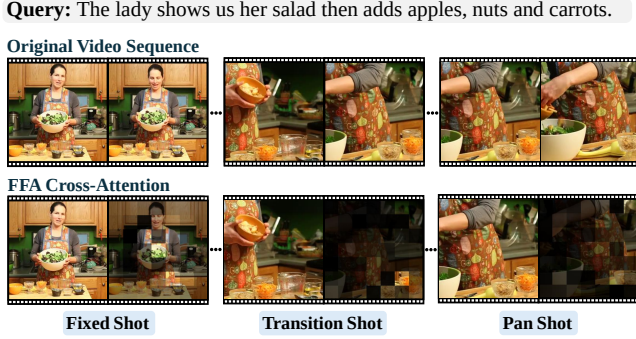


Figure 7: Visualization of FFA temporal cross-attention mechanism across different shot types in video sequences.

TP module, built upon TAME, further enhances retrieval performance through its prototype selection mechanism, as evidenced by the performance gap when removing TP alone. The $\mathcal{L}_{CT}$ effectively captures task-wise embedding overlaps, demonstrated by consistent R@1 decreases across datasets when removed.

Analysis of FFA Temporal Cross-attention Mechanism. Figure 7 visualizes how FFA cross-attention works in video sequences across different types of camera shots. In fixed shot (left), where frames remain mostly static, FFA maintains balanced attention distribution despite changes in object actions, effectively capturing inter-frame relationships. In transition shot (middle), where scene composition changes significantly between frames, FFA’s attention mechanism focuses on identifying visual consistencies (e.g., carrot bowl) to maintain continuity of shared objects and backgrounds. In pan shot (right), where the camera moves upward-right, FFA maintains attention between corresponding regions across frames, with stronger attention weights on overlapping elements between consecutive frames.

5.3 Hyper-parameter Study (RQ3)

We conduct further analysis to understand how key hyper-parameters impact our model’s performance across all datasets in Figure 8.

Number of FFA adapters: We investigate the impact of FFA adapters by implementing them starting from the shallowest layers of the CLIP Transformer blocks. As illustrated in Figure 8, we experiment with 4 to 12 (covering all blocks) FFA adapters across various CL settings. The results demonstrate consistent performance trends within each dataset, with 10-12 FFA adapters emerging as the optimal range, highlighting the FFA adapter’s effectiveness in learning video temporal information.

Number of experts used in TAME: We analyze the effect of varying the number of experts in TAME from 1 to 20. Experimental results demonstrate that model performance does not scale linearly with the number of experts. Our analysis reveals that configurations with 5 and 10 experts achieve the optimal trade-off between model effectiveness and computational overhead.

Scales of CTL ($\mathcal{L}_{CT}$): We investigate the impact of scaling the CTL ($\mathcal{L}_{CT}$) component in the overall objective function. As shown in Figure 8, adjusting the scaling coefficient of the $\mathcal{L}_{CT}$ demonstrated marginal impact on the model’s final R@1. Through evaluation of different scaling factors, we find that setting the $\mathcal{L}_{CT}$ coefficient to 0.6 achieved optimal performance by effectively balancing the

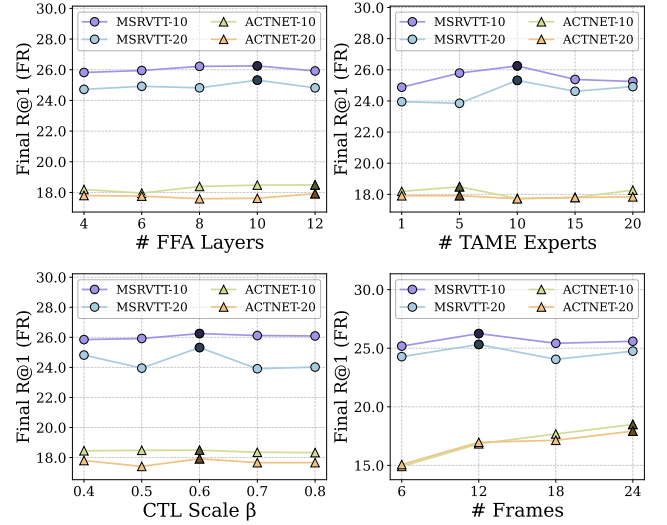


Figure 8: Hyper-parameter sensitivity on two datasets with 10 and 20 tasks respectively.

contributions of in-task and cross-task contrastive objectives.

Number of sampled frames: We conduct experiments on the number of sampled frames per video during both training and inference phases. As shown in Figure 8, in MSRVTT dataset, the model performance does not consistently improve with increased frames. Specifically, undersampling at 6 frames likely leads to performance degradation due to loss of critical temporal information, while excessive frame sampling introduces noise that adversely affects the quality of video representations. In contrast, for ACTNET dataset, as specified in section 4.1, since its videos are approximately five times longer than those in MSRVTT, increasing the number of sampled frames leads to performance improvements by better capturing the extended temporal dynamics.

6 Conclusion

In this paper, we introduce the novel research problem of Continual Text-to-Video Retrieval (CTVR), addressing the challenges posed by the dynamic and evolving nature of video content. To tackle the limitations of existing TVR and CL approaches, we propose FrameFusionMoE to maintain model plasticity while mitigating catastrophic forgetting. FrameFusionMoE incorporates the Frame Fusion Adapter (FFA) for capturing video temporal dynamics, and the Task-Aware Mixture-of-Experts (TAME) for ensuring consistent query-video alignment across tasks through task-aware routing. Comprehensive experiments demonstrate that FrameFusionMoE consistently outperforms existing methods. By establishing a benchmark for CTVR and providing a detailed analysis of current state-of-the-art methods, our work paves the way for future research in this emerging field.

Acknowledgments

This work is supported by the Australian Research Council under the streams of Discovery Project (No. DP240101814), Discovery Early Career Research Award (No. DE230101033), Linkage Project (No. LP230200892), and Centre of Excellence (No. CE200100025).

References

- [1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. 2018. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*. 139–154.
- [2] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Nieves. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 961–970.
- [3] Haoran Chen, Zuxuan Wu, Xintong Han, Menglin Jia, and Yu-Gang Jiang. 2025. Promptfusion: Decoupling stability and plasticity for continual learning. In *European Conference on Computer Vision*. Springer, 196–212.
- [4] Zhi Chen, Jingjing Li, Yadan Luo, Zi Huang, and Yang Yang. 2020. Canzsl: Cycle-consistent adversarial networks for zero-shot learning from natural language. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 874–883.
- [5] Zhi Chen, Yadan Luo, Ruihong Qiu, Sen Wang, Zi Huang, Jingjing Li, and Zheng Zhang. 2021. Semantics disentangling for generalized zero-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 8712–8720.
- [6] Zhi Chen, Sen Wang, Jingjing Li, and Zi Huang. 2020. Rethinking generative zero-shot learning: An ensemble learning perspective for recognising visual patches. In *Proceedings of the 28th ACM international conference on multimedia*. 3413–3421.
- [7] Zhi Chen, Zecheng Zhao, Yadan Luo, and Zi Huang. 2024. FastEdit: Fast Text-Guided Single-Image Editing via Semantic-Aware Diffusion Fine-Tuning. *arXiv preprint arXiv:2408.03355* (2024).
- [8] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 191–198.
- [9] Chaorui Deng, Qi Chen, Pengda Qin, Da Chen, and Qi Wu. 2023. Prompt switch: Efficient clip adaptation for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15648–15658.
- [10] Yuxuan Ding, Lingqiao Liu, Chunna Tian, Jingyuan Yang, and Haoxuan Ding. 2022. Don't stop learning: Towards continual learning for the clip model. *arXiv preprint arXiv:2207.09248* (2022).
- [11] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).
- [12] Bo Fang, Wenhao Wu, Chang Liu, Yu Zhou, Yuxin Song, Weiping Wang, Xiangbo Shu, Xiangyang Ji, and Jingdong Wang. 2023. Uatvr: Uncertainty-adaptive text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13723–13733.
- [13] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. 2021. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097* (2021).
- [14] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. 2021. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097* (2021).
- [15] William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research* 23 (2022), 1–39.
- [16] Robert M French. 1999. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences* (1999), 128–135.
- [17] Saurabh Garg, Mehrdad Farajtabar, Hadi Pouransari, Raviteja Vemulapalli, Sachin Mehta, Oncel Tuzel, Vaishaal Shankar, and Fartash Faghri. 2023. Tic-clip: Continual training of clip models. *arXiv preprint arXiv:2310.16226* (2023).
- [18] Satya Krishna Gorti, Noël Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. 2022. X-pool: Cross-modal language-video attention for text-video retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5006–5015.
- [19] Peiyan Guan, Renjing Pei, Bin Shao, Jianzhuang Liu, Weimian Li, Jiayi Gu, Hang Xu, Songcen Xu, Youliang Yan, and Edmund Y Lam. 2023. Pidro: Parallel isomeric attention with dynamic routing for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11164–11173.
- [20] Lei Guo, Hongzhi Yin, Qinyong Wang, Tong Chen, Alexander Zhou, and Nguyen Quoc Viet Hung. 2019. Streaming Session-based Recommendation. In *KDD, ACM*. 1569–1577.
- [21] Lokesh Gupta. 2024. How many daily searches are made on YouTube? LinkedIn Pulse. <https://www.linkedin.com/pulse/how-many-daily-searches-made-youtube-lokesh-gupta-m6or/>
- [22] Raia Hadsell, Dushyant Rao, Andrei A Rusu, and Razvan Pascanu. 2020. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences* 24, 12 (2020), 1028–1040.
- [23] Feng He, Qi Wang, Zhifan Feng, Wenbin Jiang, Yajuan Lü, Yong Zhu, and Xiao Tan. 2021. Improving video retrieval by adaptive margin. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 1359–1368.
- [24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.
- [25] YenChang Hsu, YenCheng Liu, Anita Ramasamy, and Zsolt Kira. 2018. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. Continual Learning Workshop. In *32nd Conference on Neural Information Processing Systems*.
- [26] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. [n. d.]. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.
- [27] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation* 3 (1991), 79–87.
- [28] Peng Jin, Hao Li, Zesen Cheng, Kehan Li, Xiangyang Ji, Chang Liu, Li Yuan, and Jie Chen. 2023. Diffusionret: Generative text-video retrieval with diffusion model. In *Proceedings of the IEEE/CVF international conference on computer vision*. 2470–2481.
- [29] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.
- [30] Dahuin Jung, Dongyoon Han, Jihwan Bang, and Hwanjun Song. 2023. Generating instance-level prompts for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11847–11857.
- [31] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114 (2017), 3521–3526.
- [32] Pandeng Li, Chen-Wei Xie, Liming Zhao, Hongtao Xie, Jiannan Ge, Yun Zheng, Deli Zhao, and Yongdong Zhang. 2023. Progressive spatio-temporal prototype matching for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4100–4110.
- [33] Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence* 40 (2017), 2935–2947.
- [34] Zhiqiu Lin, Jia Shi, Deepak Pathak, and Deva Ramanan. 2021. The clear benchmark: Continual learning on real-world imagery. In *Thirty-fifth conference on neural information processing systems datasets and benchmarks track (round 2)*.
- [35] Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. 2022. Ts2-net: Token shift and selection transformer for text-video retrieval. In *European conference on computer vision*. 319–335.
- [36] Vincenzo Lomonaco and Davide Maltoni. 2017. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on robot learning*. PMLR, 17–26.
- [37] David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. *Advances in neural information processing systems* 30 (2017).
- [38] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2022. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing* 508 (2022), 293–304.
- [39] Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. 2023. Understanding plasticity in neural networks. In *International Conference on Machine Learning*. PMLR, 23190–23211.
- [40] James L McClelland, Bruce L McNaughton, and Randall C O'Reilly. 1995. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review* (1995), 419.
- [41] Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*. Elsevier, 109–165.
- [42] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. 2024. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems* 36 (2024).
- [43] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*. 2630–2640.
- [44] Julien Nicolas, Florent Chieroni, Imtiaz Ziko, Ola Ahmad, Christian Desrosiers, and Jose Dolz. 2023. MoP-CLIP: A mixture of prompt-tuned CLIP models for domain incremental learning. *arXiv preprint arXiv:2307.05707* (2023).
- [45] Ruihong Qiu, Hongzhi Yin, Zi Huang, and Tong Chen. 2020. GAG: Global Attributed Graph Neural Network for Streaming Session-based Recommendation. In *SIGIR, ACM*. 669–678.
- [46] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. 8748–8763.
- [47] Sylvester-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2001–2010.
- [48] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesaro. 2018. Learning to learn without forgetting by maximizing transfer and minimizing interference. *arXiv preprint arXiv:1810.11910* (2018).
- [49] Ryne Roady, Tyler L Hayes, Hitesh Vaidya, and Christopher Kanan. 2020. Stream-51: Streaming classification and novelty detection from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

- 228–229.
- [50] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarsz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* (2017).
 - [51] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. 2017. Continual learning with deep generative replay. *Advances in neural information processing systems* 30 (2017).
 - [52] Tejas Srinivasan, Ting-Yun Chang, Leticia Pinto Alva, Georgios Chochlakis, Mohammad Rostami, and Jesse Thomason. 2022. Climb: A continual learning benchmark for vision-and-language tasks. *Advances in Neural Information Processing Systems* 35 (2022), 29440–29453.
 - [53] Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Chengzhong Xu. 2024. HydraLoRA: An Asymmetric LoRA Architecture for Efficient Fine-Tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*.
 - [54] Hao Xue Celso De Melo Flora Salim Tianqi Tang, Shohreh Deldari. 2024. ViLCo-Bench: Video Language COntinual learning Benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
 - [55] Guido M Van de Ven and Andreas S Tolias. 2019. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734* (2019).
 - [56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
 - [57] Andrés Villa, Kumail Alhamoud, Victor Escorcia, Fabian Caba, Juan León Alcázar, and Bernard Ghanem. 2022. vclimb: A novel video class incremental learning benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 19035–19044.
 - [58] Andrés Villa, Juan León Alcázar, Motasem Alfarra, Kumail Alhamoud, Julio Hurtado, Fabian Caba Heilbron, Alvaro Soto, and Bernard Ghanem. 2023. PIVOT: Prompting for Video Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
 - [59] Jiamian Wang, Guohao Sun, Pichao Wang, Dongfang Liu, Sohail Dianat, Majid Rabbani, Raghuvver Rao, and Zhiqiang Tao. 2024. Text Is MASS: Modeling as Stochastic Embedding for Text-Video Retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16551–16560.
 - [60] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. 2022. S-Prompts Learning with Pre-trained Transformers: An Occam's Razor for Domain Incremental Learning. In *Conference on Neural Information Processing Systems (NeurIPS)*.
 - [61] Yabin Wang, Zhiheng Ma, Zhiwu Huang, Yaowei Wang, Zhou Su, and Xiaopeng Hong. 2023. Isolation and impartial aggregation: A paradigm of incremental learning without interference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 10209–10217.
 - [62] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. 2022. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*. 631–648.
 - [63] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 139–149.
 - [64] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. 2022. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7959–7971.
 - [65] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. 2023. Cap4video: What can auxiliary captions do for text-video retrieval?. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10704–10713.
 - [66] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. 2023. Cap4video: What can auxiliary captions do for text-video retrieval?. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10704–10713.
 - [67] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5288–5296.
 - [68] Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. 2023. Clip-vip: Adapting pre-trained image-text model to video-language alignment. In *The Eleventh International Conference on Learning Representations*.
 - [69] Prateek Yadav, Qing Sun, Hantian Ding, Xiaopeng Li, Dejiao Zhang, Ming Tan, Xiaofei Ma, Parminder Bhatia, Ramesh Nallapati, Murali Krishna Ramanathan, et al. 2023. Exploring continual learning for code generation models. *arXiv preprint arXiv:2307.02435* (2023).
 - [70] Xiangpeng Yang, Linchao Zhu, Xiaohan Wang, and Yi Yang. 2024. DGL: Dynamic Global-Local Prompt Tuning for Text-Video Retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 6540–6548.
 - [71] Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. 2024. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 23219–23230.
 - [72] Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European conference on computer vision (ECCV)*. 471–487.
 - [73] Wei Yuan, Hongzhi Yin, Tiek He, Tong Chen, Qiufeng Wang, and Lizhen Cui. 2022. Unified Question Generation with Continual Lifelong Learning. In *WWW*. ACM, 871–881.
 - [74] Friedemann Zenke, Ben Poole, and Surya Ganguli. 2017. Continual learning through synaptic intelligence. In *International conference on machine learning*. 3987–3995.
 - [75] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. 2023. 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*. 19148–19158.
 - [76] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. 2022. Centerclip: Token clustering for efficient text-video retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 970–981.
 - [77] Zangwei Zheng, Mingyuan Ma, Kai Wang, Ziheng Qin, Xiangyu Yue, and Yang You. 2023. Preventing zero-shot transfer degradation in continual learning of vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 19125–19136.
 - [78] Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. 2024. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision* (2024), 1–21.
 - [79] Da-Wei Zhou, Yuanhan Zhang, Jingyi Ning, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. 2023. Learning without forgetting for vision-language models. *arXiv preprint arXiv:2305.19270* (2023).