This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

AdaCM^2: On Understanding Extremely Long-Term Video with Adaptive Cross-Modality Memory Reduction

Yuanbin Man[1], Ying Huang[1], Chengming Zhang[2], Bingzhe Li[3], Wei Niu[4], Miao Yin[1†]
[1]Department of CSE, University of Texas at Arlington, [2]Department of CS, University of Houston,
[3]Department of CS, University of Texas at Dallas, [4]School of Computing, University of Georgia
{yuanbin.man, ying.huang}@uta.edu, czhang48@uh.edu,
bingzhe.li@utdallas.edu, wniu@uga.edu, miao.yin@uta.edu

Abstract

The advancements in large language models (LLMs) have propelled the improvement of video understanding tasks by incorporating LLMs with visual models. However, most existing LLM-based models (e.g., VideoLLaMA, VideoChat) are constrained to processing short-duration videos. Recent attempts to understand long-term videos by extracting and compressing visual features into a fixed memory size. Nevertheless, those methods leverage only visual modality to merge video tokens and overlook the correlation between visual and textual queries, leading to difficulties in effectively handling complex question-answering tasks. To address the challenges of long videos and complex prompts, we propose AdaCM^2, which, for the first time, introduces an adaptive cross-modality memory reduction approach to video-text alignment in an auto-regressive manner on video streams. Our extensive experiments on various video understanding tasks, such as video captioning, video question answering, and video classification, demonstrate that AdaCM^2 achieves state-of-the-art performance across multiple datasets while significantly reducing memory usage. Notably, it achieves a 4.5% improvement across multiple tasks in the LVU dataset with a GPU memory consumption reduction of up to 65%.

1. Introduction

Video understanding is an important task in computer vision and artificial intelligence, which involves processing and reasoning over visual and textual information. While the recent success of large language models (LLMs) [5, 31, 34, 39] has significantly improved video-language models [27, 29], prior work has primarily focused on short video understanding tasks, typically with videos ranging from 5 to 15 seconds. However, long-term video understanding [45], a sub-technique that develops models to process richer information, has played a crucial role in real-world applications such as movie analysis and video retrieval. Unfortunately, it poses significant challenges as video length increases, especially the *large memory consumption* challenge. The number of frames the model must process grows rapidly, leading to substantial memory consumption, thereby preventing prior approaches from processing such long videos.

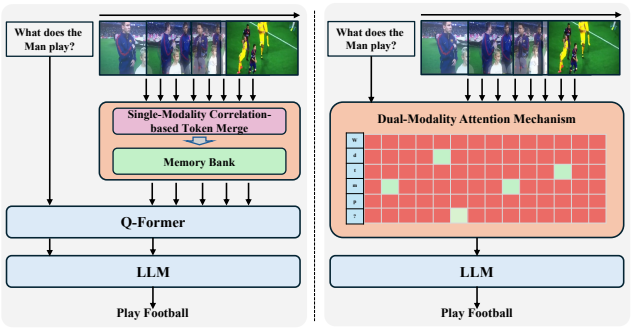


Figure 1. (Left) Existing approaches compress visual features of videos via single-modality correlation; (Right) Our **AdaCM^2** reduces video memory adaptively based on cross-modality attention.

To solve the *large memory consumption* challenge, many approaches focus on compressing video tokens. For instance, MA-LMM [16] employs a memory bank to compress visual tokens based on the cosine similarities of adjacent two frames. Koala [37] passes multiple segments of video into tokenizer functions that aggregate visual tokens to handle long videos. Even though those methods reduce memory consumption, they still suffer from two significant limitations. 1) *Ignoring text-driven information:* As shown in Figure 1, existing works compress visual information without considering textual information, leading to the loss of vital visual tokens that are highly related to

[†]Corresponding author.

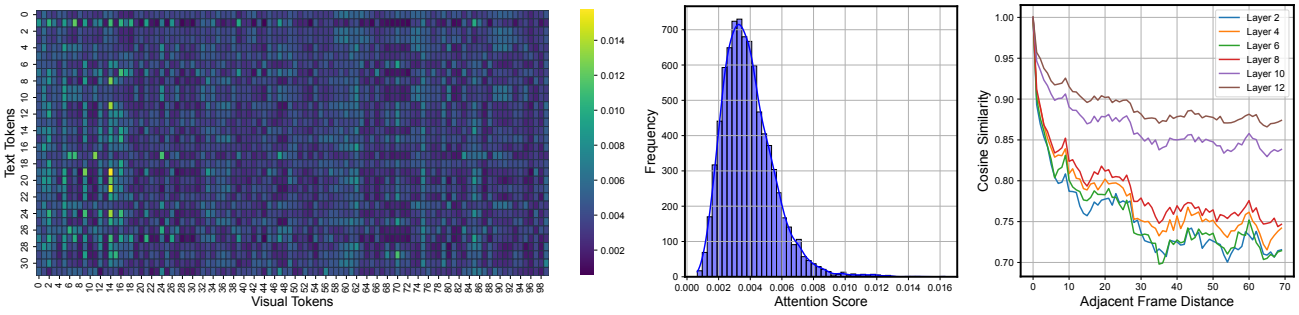


Figure 2. The case study of AdaCM^2 **zero-shot** on Ego4D[15] dataset. As shown, AdaCM^2 can 1) summarize an extremely long video lasting **over 2 hours** with limited memory consumption and identify the number on a person's back at the end accurately, 2) answer questions related to a mid-length video spanning **more than 20 minutes**.

the text prompt, particularly in Visual Question Answering (VQA) [2, 12, 20, 50] tasks with complex text prompts. 2) *The lack of similarity adaptability:* Previous methods attempt to compress visual tokens within the fixed and predefined frame interval, failing to capture the dynamic nature of video information and leading to inefficient and inflexible memory reduction. As shown in Figure 3c, the similarities of adjacent frames vary across different layers. Moreover, temporally distant frames still exhibit high similarity in deep layers. Consequently, if not appropriately addressed, such limitations will severely hinder their performance in many practical applications.

To address the above limitations, in this paper, we propose AdaCM^2, on understanding extremely long-term video with adaptive cross-modality memory reduction. As illustrated in Figure 1, our key idea is to *adaptively preserve a certain number of crucial visual tokens that are most relevant to text queries across different layers based on cross-modality attention*. We introduce two intriguing observations: 1) Only a subset of visual key states exhibit high correlations with text query; 2) Correlation varies across different layers. Our observations open opportunities for effective memory reduction by fully exploiting the inherent cross-modal capabilities of the existing visual encoders to compress visual tokens. Specifically, we first extract visual representations from video frames using a frozen visual encoder, i.e., Q-Former [23]. Then, to solve the substantial memory consumption challenge and model long-term temporal connection, we propose AdaCM^2 attention to learn the "query" regressively in a frame-by-frame manner within the Q-Former. AdaCM^2 attention mechanism enables different numbers of visual tokens preserved across different layers based on cross-modality correlation. Finally, the learned queries are sent to LLM to generate the answer. To the best of our knowledge, AdaCM^2 is *the first* framework that leverages cross-modal attention integrated into the visual encoder to enable extremely long-term video understanding with efficient memory reduction. Moreover, our AdaCM^2 can improve BLIP-based [11, 22, 23] models in a plug-and-play manner, enhancing their capability to process long-term video effectively. Our core contributions are summarized as follows:

- We propose an efficient video memory reduction framework, AdaCM^2 for long-term video understanding based on cross-modality attention, which, for the first time, facilitates interaction between visual features and text prompts in video understanding.
- We present a cross-modality attention module that adaptively analyzes and evaluates visual tokens according to the correlation with input text prompts, enabling extremely long-term video tasks with a dynamic crucial video token-preserving strategy.
- We conduct experiments on multiple video understanding tasks. Our proposed AdaCM^2 achieves 4.5% accuracy improvement across multiple tasks in the LVU [45] dataset. Particularly, AdaCM^2 exhibits promising performance on the VQA and video captioning tasks. More importantly, AdaCM^2 reduces GPU memory consumption by 65%, highlighting superior performance and efficiency in practice.



(a) (b) (c)

Figure 3. Visualization for cross-modality attention, generated using a randomly sampled video from the MSR-VTT [48] dataset. (a) Cross-attention score map of the 74th frame in the final layer and last head. (b) Cross-attention score distribution of the 80th frame in the final layer and last head. (c) The layer-wise cosine similarities of attention scores between the current frame and adjacent frames.

2. Related Work

Video-Language Models. With the recent advancements of large language models (LLMs) [5, 31, 34, 39], video-language models have been integrated by LLMs with image encoders for multi-modal understanding and reasoning [3, 7, 30]. BLIP-2 [23] introduces a lightweight querying transformer [23] to bridge the modality gap between the frozen pre-trained image encoder and LLMs. InstructBLIP [11] further extracts informative visual features tailored to the given instruction with instruction-aware Query Transformer. LLaVA [53] leverages language-model to generate multi-modal instruction-following data and improves model generalization ability. VisionLLM [7] provides a unified perspective for vision and language tasks by treating images as a foreign language. It aligns vision-centric tasks with language tasks, which can be flexibly defined and managed using language instructions. However, these models are prone to significant memory overhead when applied to long video understanding tasks.

Long-Term Video Understanding. Long video understanding focuses on detecting long-range patterns in videos longer than 30 seconds or even several minutes. To reduce memory and computational requirements, [16, 35] reduce the redundancy of visual information based on the cosine similarities of adjacent visual tokens. Other works like Vis4mer [19] leverage a standard transformer encoder for short-range spatiotemporal feature extraction and a multi-scale temporal Selective Structured State-Spaces (S4) [42] decoder for long-range temporal reasoning. Koala [37] splits a long video into multiple segments and then aggregates visual tokens to process long videos. Considering that the goal of long-term video understanding is to answer the text question corresponding to the video, our method considers the correlation between the visual features and the input text information based on adaptive cross-modality attention, significantly reducing memory consumption and enabling extremely long-term video understanding.

KV Cache Eviction. KV cache eviction, a memory reduction method that retains important context key-value pairs, is widely adopted in LLMs inference. H_2O [54] finds that keeping the recent tokens, together with "heavy-hitter" tokens measured by attention scores, is sufficient to maintain LLM's performance. Similarly, KeyFormer [1] retains only the key tokens in the KV cache by identifying these crucial tokens through a novel scoring function. In addition, based on the observation that KV cache states are highly similar between adjacent layers in the middle-to-deep sections of LLM, MiniCache [26] compresses the KV cache across layers using a novel depth-based approach, significantly reducing the memory footprint for LLM inference. Building on the success of the eviction-based method in managing long-context LLMs, we efficiently process long-term videos based on a similar philosophy. Moreover, our method focuses on reducing the redundancy of visual tokens in the long video understanding task, which is more challenging due to more information contained in videos and the interactions between visual and text modalities.

3. Observations

To investigate the main bottleneck that hinders long-term video understanding, we conduct a comprehensive study on the process between video frames and text prompts, which generally perform in the Q-Former. This process aligns visual and textual information and causes substantial memory consumption when video lengths increase. In this section, we will first analyze the cross-attention sparsity within a frame and then show the generalization of cross-attention sparsity across videos and layers. These observations demonstrate the redundancy in the dual-modality processing and inspire the foundation of our approach AdaCM^2 proposed in Section 4.

3.1. Intra-Frame Cross-Attention Sparsity

Existing approaches compress visual tokens solely based on the similarities among video frames. However, those ap-

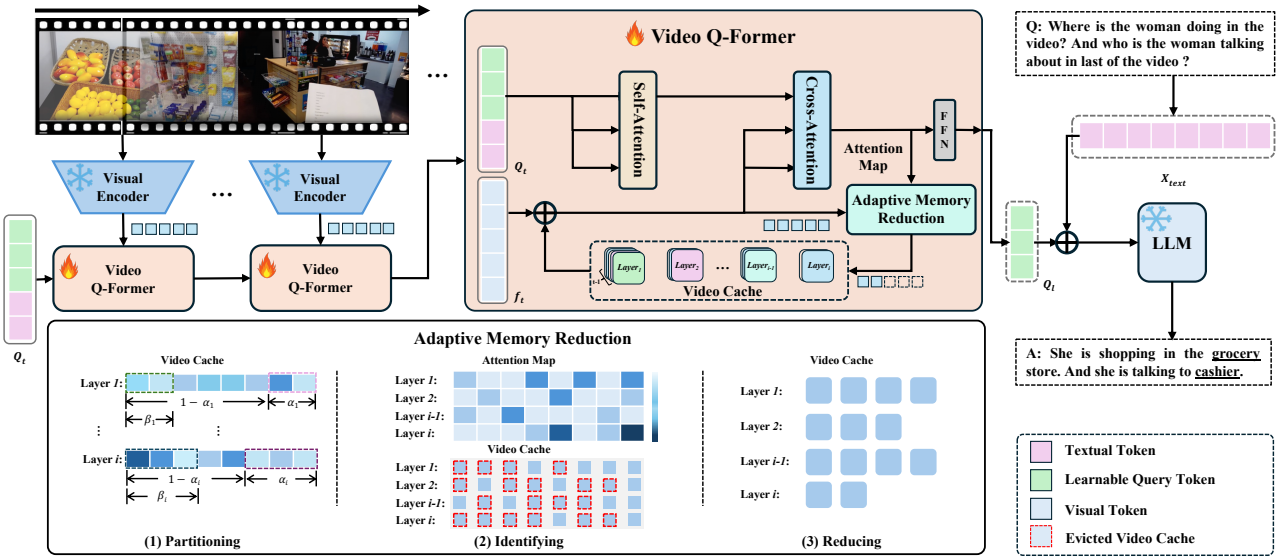


Figure 4. The framework of AdaCM^2. With video and text query as input, AdaCM^2 first utilizes a visual encoder to extract visual features from video frames. Then, video Q-Former embeds the correlation between visual features and the text prompt into a learnable query in a regressive manner. Finally, LLM generates the answer based on the length-limited query embedding. To reduce memory consumption challenge during the process of *Adaptive Memory Reduction*, the *Video Cache* is partitioned into previous and recent parts. Based on cross-modality attention score, AdaCM^2 then identifies important visual features and removes layer-wise unimportant visual tokens from cache. The snowflake denotes frozen pre-trained models, while the fire tag represents models that are fine-tuned.

proaches may miss key visual tokens highly related to textual input, leading to accuracy loss for tasks with complex text questions. Motivated by [14, 46], which have shown that retaining only a subset of salient tokens responsible for the majority of attention scores can be sufficient to maintain the performance of LLMs, we observe the similar cross-attention sparsity in the visual-textual alignment process.

Observation 1: Only a subset of visual tokens exhibits high correlations to the text query within a frame. In Figure 3a, we visualize the cross-attention scores for the 74-th frame in the final layer and the last attention head by performing inference on Q-Former with randomly sampling data from the MSR-VTT [48] dataset. The dark color indicates that the visual token exhibits a slight cross-modality correlation with the text token. We can observe that only a subset of visual tokens within a frame exhibits high correlations to text tokens. Moreover, the cross-attention scores exhibit a normal distribution as depicted in Figure 3b, with significant correlations primarily observed in the tail values. This observation motivates us to develop an algorithm that identifies necessary visual tokens according to cross-modality attention scores, thereby preserving crucial visual information most important to the textual prompts.

3.2. Layer-Wise Cross-Attention Similarity

We have demonstrated the cross-attention sparsity within a video frame. Due to the significantly increased frame length, a long-term video contains a substantial number of visual tokens. These tokens are grouped and undergo multi-layer cross-attention operations with prompt tokens. To reduce the memory consumption of redundant visual tokens globally, we further analyze cross-attention sparsity across frames and layers.

Observation 2: Correlation varies across different layers. Based on the BLIP2 [23] Model, we conduct zero-shot inference across multiple datasets and tasks. Figure 3c reveals that the cross-attention scores exhibit high similarity between adjacent frames, where cosine similarity exceeds 90% among the recent five frames. More importantly, the similarity is more significant in deeper layers than in shallow layers. This observation highlights the consistent redundancy throughout the video and the diverse levels of redundancy across different layers. Correspondingly, we propose to *adaptively* reduce visual memory consumption across different layers according to layer-wise cross-modality attention, making memory compression dynamic and flexible.

4. Methodology: AdaCM^2

We present AdaCM^2, an adaptive cross-modality memory reduction framework for extremely long-term video understanding. AdaCM^2 consists of three stages, including 1) video feature extraction by a visual encoder; 2) adaptive memory reduction based on cross-modality attention with visual-textual embedding alignment; 3) text generation with a large language model, as illustrated in Figure 4.

AdaCM^2 adaptively reduces peak memory consumption by regressively generating learnable query tokens that preserve the temporal continuity of video and a layer-wise cross-modality memory reduction algorithm.

4.1. Video Feature Extraction

As shown in Figure 4, similar to common video understanding workflow [23], AdaCM^2 extracts video features using a pre-trained visual encoder. Given a video with a sequence of T frames, the encoder first encodes each frame and generates the corresponding video features $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots, \boldsymbol{x}_T]$, where $\boldsymbol{x}_t \in \mathbb{R}^{P\times C}$ is the frame features at time t, P and C denote the number of visual tokens about each frame and the channel dimension of each token, respectively. Then, to incorporate temporal information into the frame-level features, a positional embedding is applied as follows:

$$\boldsymbol{f}_t = \boldsymbol{x}_t + \mathcal{E}(t), \boldsymbol{f}_t \in \mathbb{R}^{P\times C}, \tag{1}$$

where $\mathcal{E}(\cdot)$ represents the position embedding of a frame, and $\boldsymbol{f}_t$ indicates the frame features with temporal information at time t.

4.2. Adaptive Memory Reduction with Cross-Modality Attention

After extracting visual features from the video, a Q-Former is leveraged to align visual and textual features. By learning a query $\boldsymbol{Q}_l \in \mathbb{R}^{N_l\times C}$ in the Q-Former model, the visual features are refined to align the text description based on multi-layers cross-modal attention mechanisms, where N_l is the number of learnable query tokens and C denotes the number of feature channels.

Visual-Textual Feature Alignment in a Regressive Manner. Unlike existing methods that directly process all frames into the Q-Former and align visual and textual information by one shot, we propose to learn the query $\boldsymbol{Q}_l$ regressively in a frame-by-frame manner, enabling the reduction of irrelevant visual tokens based on the cross-modality correlation in a limited memory size.

AdaCM^2 utilizes *video cache* and current visual features to align text features. To be specific, let $\boldsymbol{K}_t \in \mathbb{R}^{tP\times C}$, $\boldsymbol{V}_t \in \mathbb{R}^{tP\times C}$ to represent *video cache* at time t, which are stored in memory by visual tokens before time t and at time t as:

$$\boldsymbol{K}_t = [\boldsymbol{K}_{\text{t-1}}, \boldsymbol{f}_t\boldsymbol{W}_K], \tag{2}$$
$$\boldsymbol{V}_t = [\boldsymbol{V}_{\text{t-1}}, \boldsymbol{f}_t\boldsymbol{W}_V], \tag{3}$$

where $\boldsymbol{W}_K$ and $\boldsymbol{W}_V$ are weight matrics. Therefore, the cross-modality attention calculation in the Q-former can be defined as

$$\boldsymbol{A}_t = \boldsymbol{S}_t \cdot \boldsymbol{V}_t = \text{softmax}\left(\frac{\boldsymbol{Q}_t \cdot \boldsymbol{K}_t^T}{\sqrt{C}}\right) \cdot \boldsymbol{V}_t, \tag{4}$$

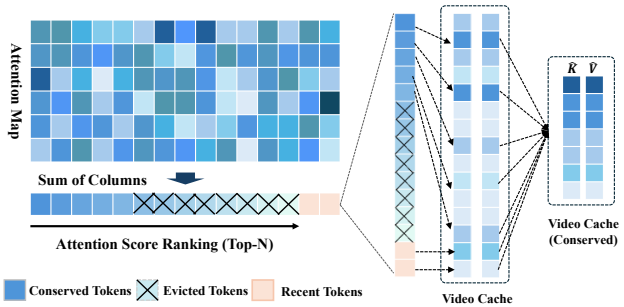


Figure 5. Illustration for our video memory reduction. The *video cache* is first partitioned into recent and previous parts. Important visual tokens with high cross-modality attention scores in the previous cache are then preserved.

where $\boldsymbol{S}_t \in \mathbb{R}^{N\times tP}$ is the attention score, $\boldsymbol{Q}_t \in \mathbb{R}^{N\times C}$ is the concatenation of the learnable query $\boldsymbol{Q}_l$ and the text embedding $\boldsymbol{X}_{\text{text}}$, N is the number of query and text tokens.

Layer-Wise Video Memory Reduction. Storing all visual tokens in the *video cache* is impossible due to the increasingly large memory cost while processing video frames, especially for long video understanding tasks. As shown in Observation 3.1, there is substantial visual redundancy in videos where only a subset of visual tokens exhibit high correlations with text tokens. Based on this observation, in AdaCM^2, we propose a cross-modality attention mechanism to reduce memory consumption according to the visual-textual redundancy. AdaCM^2 first identifies important visual features based on cross-modality attention score and then adaptively reduces video memory by removing layer-wise visual tokens not less correlated to textual information.

1 Identifying Important Visual Features based on Cross-Modality Attention Score. For a video token $\boldsymbol{f}_t(i)$ at spatial location $i \in \{1, \cdots, tP\}$, the cross-modality attention score $\boldsymbol{S}_t^c(i)$ accumulates the attention scores between $\boldsymbol{f}_t(i)$ and text tokens, which can be formulated as

$$\boldsymbol{S}_t^c(i) = \sum_{j=1}^{j=N} \boldsymbol{S}_t(j, i). \tag{5}$$

Cross-modality attention scores capture the correlation between visual and textual features. Therefore, we identify a video token as pivotal if it receives a high cross-modality attention score.

2 Adaptively Reducing Video Cache Layer-Wise. As demonstrated in Observation 3.2, visual features exhibit different levels of similarity with textual features across different layers. Guided by cross-modal attention score, we further design an adaptive video memory reduction algorithm for the *video cache*, as shown in Figure 5.

According to the order that visual token stored in the *video cache*, at time t, we split the $\boldsymbol{K}_t$ for the current layer

into *previous cache*, $\hat{\boldsymbol{K}}_t$, and *recent cache*, $\tilde{\boldsymbol{K}}_t$:

$$\begin{aligned} \boldsymbol{K}_t &= [\hat{\boldsymbol{K}}_t, \tilde{\boldsymbol{K}}_t], \\ |\hat{\boldsymbol{K}}_t|/|\tilde{\boldsymbol{K}}_t| &= (1-\alpha)/\alpha \end{aligned} \tag{6}$$

where $|\cdot|$ measures the number of tokens in cache, α is the split ratio. For the visual tokens in *recent cache*, $\tilde{\boldsymbol{K}}_t$, that reserves the latest information, we still keep it in the memory. On the other hand, for the *previous cache*, $\hat{\boldsymbol{K}}_t$, we only retain visual tokens with top-β cross-modality attention scores in memory and thus obtain the conserved cache $\overline{\boldsymbol{K}}_t$. The process is defined as follows:

$$\begin{aligned} \overline{\boldsymbol{K}}_t &= \left\{ \hat{\boldsymbol{K}}_t(i) \mid i \in \text{argtop}(\hat{\boldsymbol{S}}_t^c, \beta) \right\}, \\ |\overline{\boldsymbol{K}}_t|/|\hat{\boldsymbol{K}}_t| &= \beta, \end{aligned} \tag{7}$$

where $\text{argtop}(\cdot, \beta)$ denotes the indices of visual tokens with top-β cross-modality attention scores, $\hat{\boldsymbol{S}}_t^c$ is the cross-modality attention scores of tokens in $\hat{\boldsymbol{K}}_t$, β is the conserve ratio. Finally, the *video cache* $\boldsymbol{K}_t$ can be compressed as $\boldsymbol{K}_t = [\overline{\boldsymbol{K}}_t, \tilde{\boldsymbol{K}}_t]$. Based on our Observation 3.2, i.e., redundancy across layers varies, our adaptive video memory reduction is performed layer-wise. Correspondingly, we set α's and β's for different layers. It is worth noting that the adaptive reduction is also applied to another type of *video cache*, $\boldsymbol{V}_t$.

Let $r = \alpha + (1-\alpha)\beta \leq 1$. Through deductive reasoning, the final size of *video cache* $\boldsymbol{K}_T$ can be derived as:

$$|\boldsymbol{K}_T| = P \sum_{t=1}^{T} r^t = \frac{Pr\left(1 - r^T\right)}{1 - r} \tag{8}$$

With the continuous increase of video duration, $T \to \infty$, $|\boldsymbol{K}_T|$ is converged to a constant $Pr/(1-r)$, showing the maximum memory requirement of AdaCM^2 to processing unbounded video length.

4.3. Text Generation

With the regressive processing of video frames, the learnable query $\boldsymbol{Q}_l \in \mathbb{R}^{N \times C}$ in Q-Former has modeled the long-term temporal connection from the input video. Then, the LLM will generate the answer based on the learned $\boldsymbol{Q}_l$, a length-limited vector aggregating the long-term input video information. Specifically, assume $\boldsymbol{Y} = \{y_1, y_2, \cdots, y_M\}$ is the answer comprising a sequence of M words, we minimize the cross-entropy loss during training as follows:

$$L(\boldsymbol{V}, \boldsymbol{P}, \boldsymbol{Y}) = -\sum_{i=1}^{M} y_i \log p(y_i | y_{<i}, \boldsymbol{Q}_l, \boldsymbol{X}_{\text{text}}) \tag{9}$$

where $p(y_i | y_{<i}, \boldsymbol{Q}_l, \boldsymbol{X}_{\text{text}})$ denotes the probability for the i-th word y_i given the preceding sequence of ground truth words $y_{<i}$.

Algorithm 1 Layer-wise Adaptive Video Reduction.

Input: Video frames $\{\boldsymbol{f}_t\}_{t=1}^{T}$, hyper-parameters α, β for each layer;
Output: Reduced video KV cache $\boldsymbol{K}_t, \boldsymbol{V}_t$.
for $t = 1$ to T **do**
  $\boldsymbol{K}_t \leftarrow [\boldsymbol{K}_{t-1}, \boldsymbol{f}_t \boldsymbol{W}_K]$; $\boldsymbol{V}_t \leftarrow [\boldsymbol{V}_{t-1}, \boldsymbol{f}_t \boldsymbol{W}_V]$;
  Partitioning: $\boldsymbol{K}_t \rightarrow [\hat{\boldsymbol{K}}_t, \tilde{\boldsymbol{K}}_t]$; $\boldsymbol{V}_t \rightarrow [\hat{\boldsymbol{V}}_t, \tilde{\boldsymbol{V}}_t]$;
  Identifying: Determine $\boldsymbol{S}_t^c$ using Eq.5;
  Reducing: Obtain $\overline{\boldsymbol{K}}_t$ using Eq.7;
  $\boldsymbol{K}_t \leftarrow [\overline{\boldsymbol{K}}_t, \tilde{\boldsymbol{K}}_t]$; $\boldsymbol{V}_t \leftarrow [\overline{\boldsymbol{V}}_t, \tilde{\boldsymbol{V}}_t]$;
end for

5. Experiments

5.1. Tasks and Datasets

Long-Term Video Understanding. We conduct experiments on three common long-term video understanding datasets, including LVU [45], Breakfast [21] and COIN[38]. We report the Top-1 accuracy as the evaluation metric. In particular, for the LVU dataset, we focus our experiments on seven classification tasks: relationship, speaking style, scene, director, genre, writer, and release year. These tasks provide a diverse range of challenges for evaluating model performance in different aspects of video understanding.

Video Question Answering. We evaluate AdaCM^{2}on two popular video question answering datasets: MSRVTT-QA [48], MSVD-QA [6]. MSRVTT-QA and MSVD-QA contain shorter videos, ranging from 10 to 15 seconds. Regarding the dataset scale, MSRVTT-QA includes 10,002 videos, MSVD-QA has 1,971 videos.

Video Captioning. We present the video captioning results with the METEOR [4] and CIDEr [40] metrics on three popular datasets, MSRVTT [48], MSVD [6], and YouCook2 [55]. It is noted that YouCook2 encompasses videos longer than five minutes, posing a significant memory consumption challenge to understand the videos across such long periods.

5.2. Implementation Details

For the visual encoder, we utilize the pre-trained image encoder ViT-G/14 [51] from EVA-CLIP[33]. To efficiently align the visual and text features, we employ the pre-trained Q-Former weights from InstructBLIP [11]. For the LLM decoding, we use the pre-trained LLM, Vicuna-7B [8] V1.1. During training, we keep the visual encoder and the LLM decoder frozen, and fine-tune the trainable parameters of Q-former. For video input, we extract frames at a speed of 10 fps. All experiments are tested on our server with eight NVIDIA RTX 6000 Ada GPUs, two AMD EPYC 9254 CPUs, and 1.5 TB memory.

Table 1. Comparison with state-of-the-art methods on the LVU dataset. The underlined number means the second best.

Model	Content			Metadata				Avg
	Relation	Speak	Scene	Director	Genre	Writer	Year	
Obj_T4mer [45]	54.8	33.2	52.9	47.7	52.7	36.3	37.8	45.1
Performer [9]	50.0	38.8	60.5	58.9	49.5	48.2	41.3	49.6
Orthoformer [32]	50.0	38.3	66.3	55.1	55.8	47.0	43.4	50.8
VideoBERT [36]	52.8	37.9	54.9	47.3	51.9	38.5	36.1	45.6
LST [19]	52.5	37.3	62.8	56.1	52.7	42.3	39.2	49.0
VIS4mer [19]	57.1	40.8	67.4	62.6	54.7	48.8	44.8	53.7
S5 [43]	**67.1**	42.1	73.5	67.3	65.4	51.3	48.0	59.2
MA-LMM [16]	58.2	**44.8**	80.3	74.6	61.0	70.4	51.9	63.0
Ours	63.1	40.2	**86.2**	**75.4**	**68.0**	**77.0**	**62.5**	**67.5**

Table 2. Comparison on the Breakfast and COIN datasets. Top-1 accuracy is reported.

Model	Breakfast	COIN
TSN [44]	-	73.4
VideoGraph [18]	69.5	-
Timeception [17]	71.3	-
GHRM [13]	75.5	-
D-Sprv. [25]	89.9	90.0
ViS4mer [19]	88.2	88.4
S5 [43]	90.7	90.8
MA-LMM [16]	93.0	93.2
Ours	**94.4**	**93.3**

Table 3. Comparison on the MSRVTT, MSVD and YouCook2 datasets, with results for the METEOR(M) and CIDEr(C) metrics.

Model	MSRVTT		MSVD		YouCook2	
	M	C	M	C	M	C
UniVL [28]	28.2	49.9	29.3	52.8	-	127.0
SwinBERT [24]	29.9	53.8	41.3	120.6	15.6	109.0
GIT [41]	32.9	73.9	51.1	180.2	17.3	129.8
mPLUG-2 [47]	**34.9**	**80.3**	48.4	165.8	-	-
VideoCoca [49]	-	73.2	-	-	-	128.0
VideoLLaMA [52]	32.9	71.6	49.8	175.3	16.5	123.7
MA-LMM [16]	33.4	74.6	51.0	179.1	**17.6**	**131.2**
Ours	33.0	73.1	**51.4**	**189.4**	**17.6**	125.6

5.3. Main Results

Long-Term Video Understanding. We compare AdaCM^2 with state-of-the-art methods on the LVU [45] benchmark. As shown in Table 1, our model outperforms the existing long-term video understanding models, including Object transformer [45], S4 [43], VIS4mer [19], VideoBERT [36], LST [19], and MA-LMM [16] across both content understanding and metadata prediction tasks. The results indicate that AdaCM^2 achieves a significant improvement across most tasks, increasing Top-1 accuracy by 4% compared to MA-LMM [16].

In Table 2, we evaluate our AdaCM^2 on the Breakfast [21] and COIN [38] datasets. It is worth noting that these datasets present a greater memory consumption challenge due to the longer and more diverse videos they contain. It is seen that our AdaCM^2 outperforms MA-LMM [16] by 1.4% and 0.1% in Top-1 accuracy, respectively. These results demonstrate that our approach achieves superior performance in long-term video understanding.

Video Captioning. Table 3 summarizes the experimental results on video captioning datasets, including MSRVTT, MSVD, and YouCook2. AdaCM^2 achieves 51.4% Meteor and 189.4% CIDEr on the MSVD datasets. The results show that AdaCM^2 outperforms the prior state-of-the-art approach, MA-LMM [16], with gains of 0.4% and 10.3%, respectively. Although mPLUG-2 has slightly better performance on MSRVTT, it demands extensive data for pre-training, leading to a training overhead.

Memory Analysis. In addition to performance evaluation, we also conduct experiments on memory usage with randomly selected videos from the LVU [45], MSRVTT [48], and MSVD [6] datasets. As shown in Figure 6, existing methods like InstructBLIP [11] and VideoLLaMA [52] instantly exhibit rapid increases in memory consumption as the number of frames increases, leading to out-of-memory (**OOM**) errors. In contrast, AdaCM^2 achieves a significant reduction in memory usage by nearly 65%, and maintains almost constant memory consumption without sacrificing performance. Furthermore, most of the occupied memory is consumed by LLM, with only a small fraction allocated for adaptive cross-modality alignment, which can be further alleviated if we use a lightweight LLM.

5.4. Ablation Studies

Memory Reduction. Our method reduces video memory adaptively based on cross-modality attention scores,

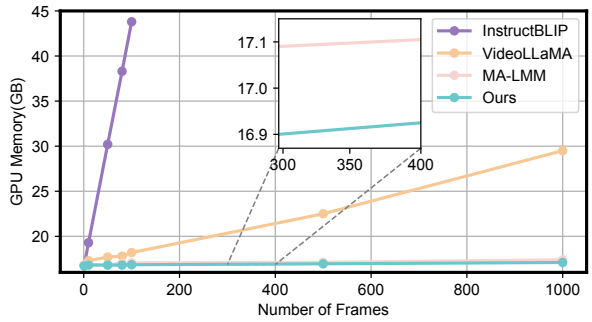


Figure 6. Practical memory consumption analysis compared to existing methods. The GPU memory usage for InstructBLIP shows an exponential increase, leading to out-of-memory (**OOM**) errors after processing 100 frames. In contrast, VideoLLaMA exhibits a linear increase in memory usage. AdaCM^2 maintains almost constant memory usage without sacrificing performance even as frames significantly increase.

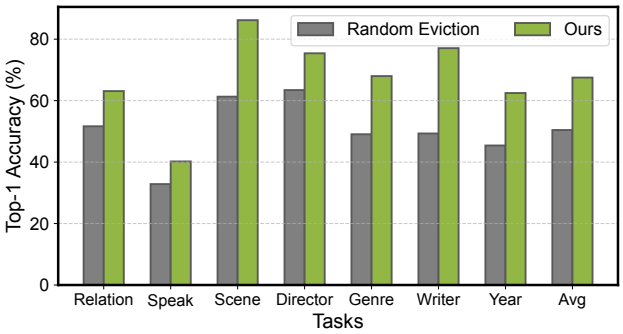
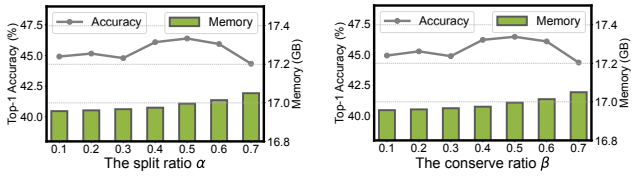


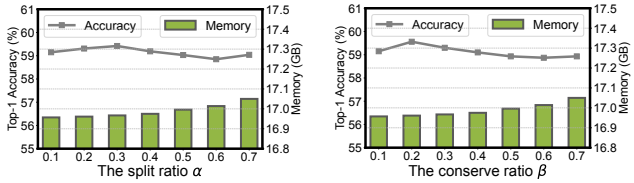
Figure 7. Performance Comparison between two memory reduction strategies, random eviction v.s AdaCM^2. Top-1 accuracy is reported here.

achieving extremely long-term video understanding with low memory cost. To investigate the importance of our memory reduction, we compare our cache strategy with the random eviction on the LVU dataset, which randomly discards the same number of visual tokens, as shown in Figure 7. It is seen that our approach significantly outperforms the random one, indicating the effectiveness of our cross-attention-based memory reduction.

Hyperparameters Analysis. Split ratio α and conserve ratio β are hyper-parameters controlling the preservation of critical visual tokens. In order to determine the appropriate settings, we explore the influence of α and β on memory reduction and performance on MSRVTT and MSVT datasets. Figure 8 shows that more retaining tokens result in higher memory usage. Moreover, with the increase of α (β), the accuracy rises to a peak and then decreases. This decrease is due to the negative impact of retaining redundant information, which diverts the model's attention away from essential tokens. To achieve an optimal balance between performance and memory efficiency, we set $\alpha = 0.1$ and $\beta = 0.1$



(a) MSRVTT-QA dataset, $\beta = 0.1$ (b) MSRVTT-QA dataset, $\alpha = 0.1$



(c) MSVD-QA dataset, $\beta = 0.1$ (d) MSVD-QA dataset, $\alpha = 0.1$

Figure 8. Influence of split ratio α and conserve ratio β on performance and memory usage.

Table 4. Ablation study of LLM decoding methods, with results reported for the METEOR(M) and CIDEr(C) metrics.

Model	MSRVTT		MSVD		YouCook2	
	M	C	M	C	M	C
FlanT5-XL	21.6↓	58.9↓	48.7↓	166.9↓	15.3↓	107.6↓
Vicuna-7B	33.0	73.1	51.4	189.4	17.6	125.6

for all layers.

LLM Decoding. LLM plays an essential role in producing instruction cues. To investigate the influence of LLMs on our AdaCM^2, we compare the results using multiple LLMs, including *FlanT5-XL* [10] and *Vicuna-7B* [8]. Table 4 indicates that Vicuna-7B achieves better performance in all tasks. Therefore, we choose the Vicuna-7B model as our LLM backbone. Since our AdaCM^2 framework is designed to accommodate various modern LLMs, we plan to conduct further tests in future work.

6. Conclusion

In this paper, we present AdaCM^2, an adaptive cross-modality memory reduction framework for extremely long-term video understanding. The key idea of AdaCM^2 is to adaptively preserve a certain number of crucial visual tokens most relevant to text queries across different layers based on cross-modality attention, addressing the substantial memory consumption and modeling long-term temporal connection. Moreover, our AdaCM^2 enables BLIP-based [11, 22, 23] models in a plug-and-play manner, enhancing their capability to process long-term video effectively. Experiments on video question understanding and captioning tasks demonstrate the superiority of the proposed AdaCM^2 over existing state-of-the-art approaches.

References

[1] Muhammad Adnan, Akhil Arunkumar, Gaurav Jain, Prashant Nair, Ilya Soloveychik, and Purushotham Kamath. Keyformer: Kv cache reduction through key tokens selection for efficient generative inference. *Proceedings of Machine Learning and Systems*, 7, 2024. 3

[2] Aishwarya Agrawal, Dhruv Batra, Marcus Rohrbach, Jiasen Lu, Michael Bernstein, and Devi Parikh. Visual question answering, 2015. 2

[3] 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. 3

[4] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, 2005. Association for Computational Linguistics. 6

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 1, 3

[6] David L. Chen and William B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL-2011)*, Portland, OR, 2011. 6, 7

[7] Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large language model for streaming video. In *CVPR*, 2024. 3

[8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 6, 8

[9] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. *CoRR*, abs/2009.14794, 2020. 7

[10] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022. 8

[11] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2, 3, 6, 7, 8

[12] Ana Claudia Akemi Matsuki de Faria, Felype de Castro Bastos, Jose Victor Nogueira Alves da Silva, Vitor Lopes Fabris, Valeska de Sousa Uchoa, Decio Gonc alves de Aguiar Neto, and Claudio Filipi Goncalves dos Santos. Visual question answering: A survey on techniques and common trends in recent literature, 2023. 2

[13] Naglaa Fathy El Dessouky and Hakeem Omar Alquaiti. Impact of green human resource management (ghrm) practices on organizational performance. In *2020 Second International Sustainability and Resilience Conference: Technology and Innovation in Building Designs(51154)*, pages 1–4, 2020. 7

[14] Suyu Ge, Yunan Zhang, Liyuan Liu, Minjia Zhang, Jiawei Han, and Jianfeng Gao. Model tells you what to discard: Adaptive KV cache compression for LLMs. In *The Twelfth International Conference on Learning Representations*, 2024. 4

[15] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 2

[16] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding. *CVPR*, 2024. 1, 3, 7

[17] Noureldien Hussein, Efstratios Gavves, and Arnold W. M. Smeulders. Timeception for complex action recognition, 2019. 7

[18] Noureldien Hussein, Efstratios Gavves, and Arnold W. M. Smeulders. Videograph: Recognizing minutes-long human activities in videos, 2019. 7

[19] Md Mohaiminul Islam and Gedas Bertasius. Long movie clip classification with state-space video models. *arXiv preprint arXiv:2204.01692*, 2022. 3, 7

[20] Vahid Kazemi and Ali Elqursh. Show, ask, attend, and answer: A strong baseline for visual question answering. *CoRR*, abs/1704.03162, 2017. 2

[21] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of Computer Vision and Pattern Recognition Conference (CVPR)*, 2014. 6, 7

[22] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2, 8

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. 2, 3, 4, 5, 8

[24] Kevin Lin, Linjie Li, Chung-Ching Lin, Faisal Ahmed, Zhe Gan, Zicheng Liu, Yumao Lu, and Lijuan Wang. Swinbert: End-to-end transformers with sparse attention for video captioning, 2022. 7

[25] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. Learning to recognize procedural activities with distant supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13853–13863, 2022. 7

[26] Akide Liu, Jing Liu, Zizheng Pan, Yefei He, Gholamreza Haffaria, and Bohan Zhuang. Minicache: Kv cache compression in depth dimension for large language models. In *NIPS*, 2024. 3

[27] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 1

[28] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation, 2020. 7

[29] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2630–2640, 2019. 1

[30] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3

[31] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022. 1, 3

[32] Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. *Advances in neural information processing systems*, 34:12493–12506, 2021. 7

[33] Ledell Wu Quan Sun, Yuxin Fang, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023. 6

[34] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. Technical report. 1, 3

[35] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232, 2024. 3

[36] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. *CoRR*, abs/1904.01766, 2019. 7

[37] Reuben Tan, Ximeng Sun, Ping Hu, Jui-Hsien Wang, Hanieh Deilamsalehy, Bryan A. Plummer, Bryan Russell, and Kate Saenko. Koala: Key frame-conditioned long video-llm. In *CVPR*, 2024. 1, 3

[38] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 6, 7

[39] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. 1, 3

[40] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726, 2014. 6

[41] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language, 2022. 7

[42] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *CVPR 2023*, 2023. 3

[43] Jue Wang, Wentao Zhu, Pichao Wang, Xiang Yu, Linda Liu, Mohamed Omar, and Raffay Hamid. Selective structured state-spaces for long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6387–6397, 2023. 7

[44] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos, 2017. 7

[45] Chao-Yuan Wu and Philipp Krähenbühl. Towards Long-Form Video Understanding. In *CVPR*, 2021. 1, 2, 6, 7

[46] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv*, 2023. 4

[47] Haiyang Xu, Qinghao Ye, Ming Yan, Yaya Shi, Jiabo Ye, Yuanhong Xu, Chenliang Li, Bin Bi, Qi Qian, Wei Wang, Guohai Xu, Ji Zhang, Songfang Huang, Fei Huang, and Jingren Zhou. mplug-2: A modularized multi-modal foundation model across text, image and video, 2023. 7

[48] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5288–5296, 2016. 3, 4, 6, 7

[49] Shen Yan, Tao Zhu, Zirui Wang, Yuan Cao, Mi Zhang, Soham Ghosh, Yonghui Wu, and Jiahui Yu. Videococa: Video-text modeling with zero-shot transfer from contrastive captioners, 2023. 7

[50] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[51] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers, 2022. 6

[52] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding, 2023. 7

[53] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 3

[54] Zhenyu Zhang, Ying Sheng, Tianyi Zhou andTianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett andZhangyang Wang, and Beidi Chen. H2o: Heavy-hitter oracle for efficient generative inference of large language models. In *NIPS*. NIPS, 2023. 3

[55] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI Conference on Artificial Intelligence*, pages 7590–7598, 2018. 6