MovieChat+: Question-aware Sparse Memory for Long Video Question Answering

Enxin Song*, Wenhao Chai*, Tian Ye, Jenq-Neng Hwang, *Life Fellow, IEEE,* Xi Li, *Senior Member, IEEE,* Gaoang Wang, *Member, IEEE*

Abstract—Recently, integrating video foundation models and large language models to build a video understanding system can overcome the limitations of specific pre-defined vision tasks. Yet, existing methods either employ complex spatial-temporal modules or rely heavily on additional perception models to extract temporal features for video understanding, and they only perform well on short videos. For long videos, the computational complexity and memory costs associated with long-term temporal connections are significantly increased, posing additional challenges. Taking advantage of the Atkinson-Shiffrin memory model, with tokens in Transformers being employed as the carriers of memory in combination with our specially designed memory mechanism, we propose MovieChat to overcome these challenges. We lift pre-trained multi-modal large language models for understanding long videos without incorporating additional trainable temporal modules, employing a zero-shot approach. Additionally, in our new version, MovieChat+, we design an enhanced vision-question matching-based memory consolidation mechanism to more significantly anchor the predictions of the visual language models in the relevant visual content. MovieChat achieves state-of-the-art performance in long video understanding, along with the released MovieChat-1K benchmark with 1K long video, 2K temporal grounding labels, and 14K manual annotations for validation of the effectiveness of our method. The code along with the dataset can be accessed via the following link.

Index Terms—Long video understanding, Vision-language model.

1 Introduction

Recent advances in Large Language Models (LLMs) [14], [20], [52], [68], [70] achieve great success in Natural Language Processing (NLP). It is a natural progression to introduce multi-modality [17] into LLMs and turn it into Multimodal Large Language Models (MLLMs), which are able to conduct multimodal rationalization and understanding. MLLMs have shown incredible emergent capabilities in various multimodal tasks such as perception (*e.g.* , existence, count, position, OCR) [1], [37], [38], [48], [75], [101], commonsense reasoning [27], [29], [37], [38], [40], [48], [67], [101], embodied agent [95]–[97], and code reasoning [22], [25], [27], [43], [45], [88], resulting in a potential path to Artificial General Intelligence (AGI). Compared to LLMs and other task-specific models, MLLMs provide a more human-like interpretation of the scenarios, a user-friendly interface for interaction, and a broader range of capabilities.

Existing vision-centric MLLMs follow the paradigm that utilizes pre-trained LLMs and visual encoders with additional learnable modules (Q-former [22], [38], [40], [92] or simple projection layer [23], [43], [48], [67]). In the video understanding field, some previous works [48], [92] follow this paradigm to build video MLLMs, while works in the other paradigm [41], [72] combine existing visual perception tools (*e.g.* , tracking and classification) and LLMs through application programming interface to build a system without training. However, these methods either employ complex spatial-temporal modules or heavily rely on additional perception tools to acquire temporal information for video understanding. Additionally, when handling long videos, the computational complexity and memory costs associated

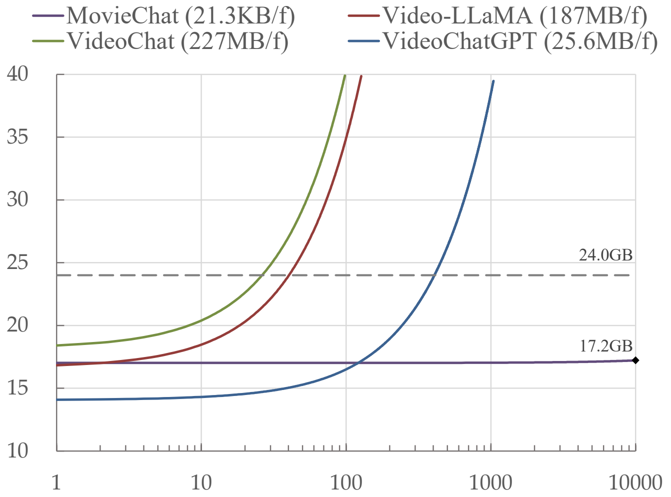


Fig. 1. Video random-access memory (VRAM) cost under gigabyte (GB) (y-axis) v.s. frame number (x-axis) comparison. We test the visual-only inference of all methods at a resolution of 224×224 without frame sampling. While the previous method can only support around 100 frames of inference, MovieChat can handle videos with $>$10K frames on a 24GB graphics card. MovieChat has a $10000\times$ advantage over other methods in terms of the average increase in VRAM cost per frame (21.3KB to ~ 200MB per frame).

- * *Equal Contributions.*
- *E. Song and G. Wang are with Zhejiang University-University of Illinois at Urbana-Champaign Institute, Zhejiang University, China.*
- *W. Chai and J.-N. Hwang are with the Department of Electrical and Computer Engineering, the University of Washington, USA.*
- *T. Ye is with Robotics and Autonomous Systems Thrust, Hong Kong University of Science and Technology (GuangZhou), China.*
- *X. Li is with the College of Computer Science and Technology, Zhejiang University, China.*

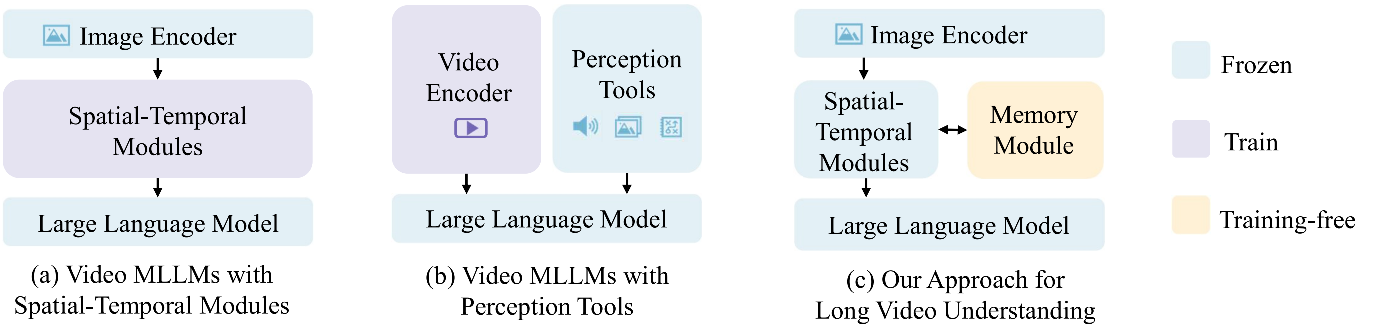


Fig. 2. Comparison between existing video MLLMs and our appraoch. While previous works either employ complex spatial-temporal modules or heavily rely on additional perception tools, and struggle with long videos, our approach is the first to address long video understanding tasks without training.

with long-term temporal connections are significantly increased, posing additional challenges, as shown in Fig. 1. Furthermore, there is also a lack of a standardized benchmark to evaluate the capabilities of these systems.

To the best of our knowledge, our work is the first to address long video understanding tasks (>10K frames). We argue that the computational complexity, memory costs, and long-term temporal connection are the main challenges for understanding long videos. Unlike other approaches that require training additional temporal modules [48], [92] in Fig. 2, our MovieChat lifts pre-trained MLLMs to understand long videos without the need for additional trainable temporal modules, employing a zero-shot approach. Inspired by the Atkinson-Shiffrin memory model [6], we propose a memory mechanism to address long video understanding tasks. This mechanism comprises a rapidly updated short-term memory and a compact, thus, sustained long-term memory. In our updated version, namely **MovieChat+**, we design a vision-question matching-based memory consolidation mechanism to enhance the compactness of memory. This mechanism significantly anchors the predictions of the visual language models in the relevant visual content. Our MovieChat+ significantly improves upon the initial version and outperforms the state-of-the-art in both short and long video question-answering tasks, surpassing even methods specifically tailored for short video question-answering challenges. As shown in Fig. 1, our approach outperforms other existing methods in terms of Video Random-Access Memory (VRAM) cost. We also release a new benchmark, MovieChat-1K, with 1K long videos and 14K manual question-answering pairs for validation of the effectiveness of our proposed method. Additionally, we expand MovieChat-1K with 2K temporal labels. The contributions of this work are summarized as follows:

- We present MovieChat, the first framework designed to support long-term videos (>10K frames), leveraging pre-trained MLLMs and employing a zero-shot, training-free memory consolidation mechanism.
- Our updated version, **MovieChat+**, enhances memory compactness through the implementation of a vision-question matching-based memory consolidation technique. This improvement significantly surpasses the initial version and outperforms the current state-of-the-art in both short and long video question-answering tasks.
- We have released the first long-video understanding benchmark, MovieChat-1K, which now includes an expansion to 2K temporal labels compared to the initial version. We conducted extensive quantitative evaluations and case studies to assess the comparable performance of both understanding capability and inference cost.

2 Related Works

2.1 Multi-modal Large Language Models

LLMs [14], [20], [52], [68], [70], [71] have achieved great success in natural language processing (NLP) tasks recently. Many works try to build MLLMs [1], [22], [27], [29], [37], [38], [47], [48], [75], [76], [88], [101] by combining models of other modalities. Flamingo [1] bridges powerful pre-trained vision-only and language-only models and achieves state-of-the-art performance with few-shot learning. BLIP-2 [38] proposes a generic and efficient pre-training strategy that bootstraps vision-language pre-training from an off-the-shelf frozen pre-trained image encoder and a frozen large language model. MiniGPT-4 [101] also aligns a frozen visual encoder with a frozen LLM, Vicuna [20], using just one projection layer to realize the system. Otter [37] showcases improved instruction-following ability and in-context learning. In the video field, ChatVideo [72] treats tracklets as the basic video unit and allows users to interact with the LLMs. VideoChat [41] integrates video foundation models and LLMs via a learnable neural interface, excelling in spatiotemporal reasoning, event localization, and causal relationship inference. Video-LLaMA [92] further leverages pre-trained models ImageBind [28] and LLaMA [70], bootstraping cross-modal training in videos following BLIP-2. Yet, these methods fail to handle long video understanding because of high computation complexity, large memory cost, and weak long-term temporal connection. Therefore, our main effort is to introduce an effective memory mechanism to overcome these challenges.

2.2 Long Video Understanding

Understanding long videos is a challenging task in computer vision. Prior arts use 3D CNN for long-term feature bank [77], object/human-centric motion [56], [78], or other

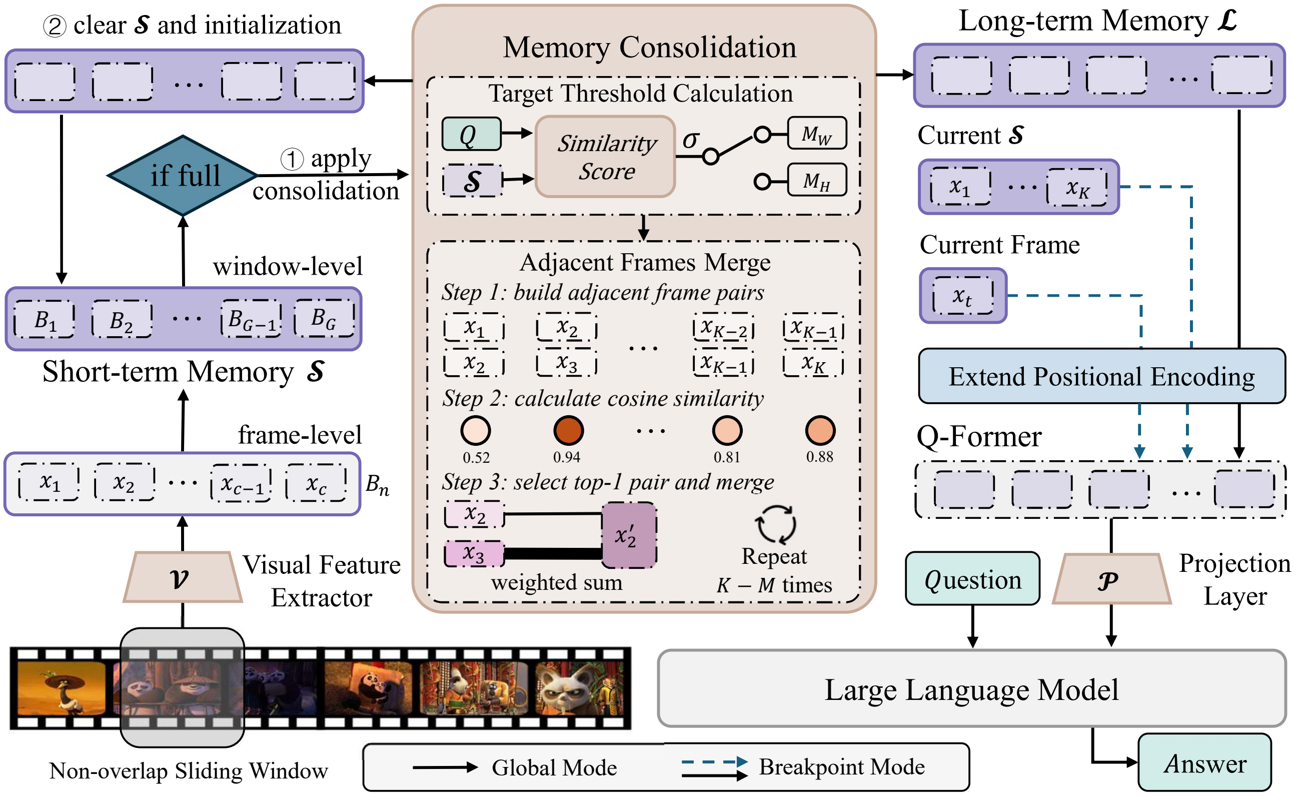

② clear $\mathcal{S}$ and initialization
if full
① apply consolidation
window-level
B_1 B_2 ⋯ B_{G-1} B_G
Short-term Memory $\mathcal{S}$
frame-level
x_1 x_2 ⋯ x_{c-1} x_c B_n
$\mathcal{V}$
Visual Feature Extractor
Non-overlap Sliding Window
Memory Consolidation
Target Threshold Calculation
Q
$\mathcal{S}$
Similarity Score
σ
M_W
M_H
Adjacent Frames Merge
Step 1: build adjacent frame pairs
x_1 x_2 ⋯ x_{K-2} x_{K-1}
x_2 x_3 x_{K-1} x_K
Step 2: calculate cosine similarity
0.52 0.94 ⋯ 0.81 0.88
Step 3: select top-1 pair and merge
x_2 x_3 x'_2
weighted sum
Repeat $K - M$ times
Long-term Memory $\mathcal{L}$
Current $\mathcal{S}$
x_1 ⋯ x_K
Current Frame
x_t
Extend Positional Encoding
Q-Former
Question
$\mathcal{P}$
Projection Layer
Large Language Model
Answer
Global Mode
Breakpoint Mode

Fig. 3. **Illustration of MovieChat, a training-free framework with question-aware memory consolidation mechanism.** MovieChat extracts video features with a sliding window and represents them in token form, which are then sequentially fed into the short-term memory frame by frame. When the fixed-length short-term memory reaches its preset limit, the earliest tokens are popped and consolidated into the long-term memory. Our approach incorporates two distinct inference modes: the global mode, which exclusively utilizes the long-term memory, and the breakpoint mode, which additionally incorporates the current short-term memory as part of the video representation. The breakpoint mode allows for understanding the video at a specific moment in time. After passing through a projection layer, the video representation is inputted into a large language model for interaction with the user.

forms [60], [79] as video representations. MIST [26] decomposes dense self-attention into a cascade segment and region selection module to increase the computation efficiency for understanding minutes of long videos. Building long-form video understanding datasets is challenging and rarely explored. [63] captures large scale data from Kinetics-400 [16], but only for generic event boundary detection tasks. [64] creates a language grounding benchmark from audio descriptions of movies, but it lacks long-term understanding evaluation. [69] successfully builds a benchmark contains multiple sources of information (*e.g.* , video clips, plots, and DVS) for question-answering tasks in the movie field. There are also several datasets of video-caption/description pairs among various domains, such as cooking (*e.g.* , MPII Cooking [57]–[59] and TACoS [54], [55]), instruction (*e.g.* , HowTo100M [50] and HiREST [90]), Ego [49], and movie (*e.g.* , MovieQA [69] and MovieNet [32]) from different sources such as YouTube [18], [50], [91], Twitter [7]–[10], and Internet [11]. Yet, those datasets lack diverse and fine-grained dense captioning for long videos.

2.3 Memory Models in Vision Tasks

There are some prior works exploring memory models [65] in various vision tasks in videos, such as video object segmentation (VOS) [19], [31], [61], [62], multi-object tracking (MOT) [2], [15], [30], [82], visual object tracking (VOT) [42], [46], [87], [100], and action understanding [73]. MeMOT [15] builds a large spatiotemporal memory that stores the past observations of the tracked objects. XMem [19] develops an architecture that incorporates multiple independent yet deeply connected feature memory storage to handle long videos with thousands of frames. Recently, [99] propose a memory streaming method to support dense video captioning. We learn from the experience of those prior arts and further adopt an effective memory mechanism in combination with LLMs.

Our method focuses on reducing the redundancy of visual tokens in the video and building a memory mechanism to pass the information among a large temporal range.

3 MovieChat

3.1 Overview

Our proposed method, MovieChat, comprises several key components, including the frame-wise visual feature extractor, the short-term memory module, the long-term memory module with question-aware consolidation strategy (our improved version, namely MovieChat+), the video projection layer, and the Large Language Model (LLM), as illustrated in Fig. 3. our approach is designed for ultra-long videos (>10K frames) understanding through interactive dialogue

Algorithm 1 Memory consolidation

```
Require: $\mathcal{S}$                                  ▷ short-term memory
  for $\mathbf{x}_i$ in $\mathcal{S}$ do
        $s^q \leftarrow sim(\mathbf{x}_i, \mathbf{q})$      ▷ question and frame similarity
  end for
  if mean($s^q$) $> \sigma$ then                          ▷ target merging threshold
        $M \leftarrow M_0$
  else
        $M \leftarrow \alpha M_0$
  end if
  while $len(\mathcal{S}) > M$ do                         ▷ iterative merge
      for $\mathbf{x}_i$ in $\mathcal{S}$ do
          $s^f \leftarrow sim(\mathbf{x}_i, \mathbf{x}_{i+1})$      ▷tokens similarity
      end for
      $m \leftarrow max(s^f)$                  ▷the maximum value index
      $\mathbf{x}_m \leftarrow merge(\mathbf{x}_m, \mathbf{x}_{m+1})$        ▷merge
      del $\mathbf{x}_{m+1}$
  end while
```

with the user. To address the impractical storage demands of concurrently storing a vast number of frames in both GPU memory and RAM, we employ a sliding window approach to efficiently process the video. The short-term memory module embeds dense tokens with sliding windows, and the long-term memory module periodically updates based on question-aware consolidation. MovieChat supports two inference modes: breakpoint mode and global mode. Breakpoint mode is used to understand a specific moment in the video, providing insights and answers based on that particular frame or scene. Global mode, on the other hand, is employed to comprehend the entire video as a whole, enabling a comprehensive understanding of the overall content and context.

3.2 Visual Feature Extraction

For visual feature extraction, instead of utilizing video-based foundational models [5], [44], we simply use an image-based model to get frame-wise features in the form of tokens. To be specific, we utilize pre-trained models as our visual feature extractor, including the ViT-G/14 from EVA-CLIP [24] and the Q-former from BLIP-2 [39]. This is mainly because 1) there are few video foundation models that make good alignment with text, and 2) our proposed memory mechanism can effectively capture temporal features. Given a raw video, the visual input $\mathbf{v} \in \mathbb{R}^{T \times 3 \times H \times W}$ is a sequence of T RGB frames of size $H \times W$ sampled from the video. The visual features are extracted in a sliding window manner:

$$B_n = \{\mathbf{x}_i = \mathcal{V}(\mathbf{v}_i) \mid \forall i = 1, ..., C\}, n = 1, ..., \lceil \frac{T}{C} \rceil, \quad (1)$$

where B_n is the n-th video clip feature within the sliding window spanning C frames. $\mathcal{V}(\cdot)$ is the visual feature extractor, taking as input a single frame $\mathbf{v}_i \in \mathbb{R}^{3 \times H \times W}$. $\mathbf{x}_i \in \mathbb{R}^{N \times D}$ denotes N extracted visual tokens with respect to each frame, and D is the feature dimension of each token.

3.3 Short-term Memory

The short-term memory maintains a fixed-length buffer to temporarily hold frame tokens. The previously extracted visual features by sliding window G times without further processing are used to construct short-term memory, which can be described by the following formulation:

$$\mathcal{S} = \bigcup_n B_n = \{\mathbf{x}_i \mid \forall i = 1, ..., K\}, n = 1, .., G, \quad (2)$$

where $\mathcal{S}$ is the short-term memory, and K is equal to $C \times G$. Note that we configure short-term memory to contain a fixed length of K frames. The design stems from the fundamental purpose of short-term memory, which is to aid in the interpretation and understanding of video content by leveraging contextual information from recent, short-term segments of the video.

As a new batch of visual tokens enters, when the short-term memory reaches its capacity, we pop the currently stored frames to the memory consolidation module and clear the short-term memory. The output video feature obtained from the consolidation module augments the long-term memory; on the other hand, it re-initializes the short-term memory with this feature. The initialization aims to communicate information between different sliding windows, thereby achieving more efficient compression.

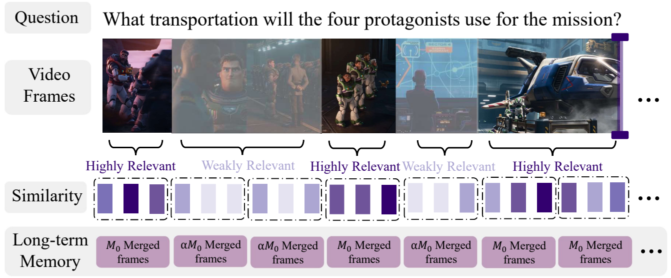


Fig. 4. Visualization of the question-aware memory consolidation strategy. If frames in the short-term memory are highly relevant to the question (indicated by dark blocks), a sparse compression strategy is implemented, compressing the frames in the short-term memory into a relatively larger number of frames. Conversely, weakly relevant segments (indicated by light blocks) undergo a dense compression strategy, compressing the frames in the short-term memory into a much smaller number of frames.

3.4 Question-aware Long-term Memory (MovieChat+)

The long-term memory can effectively avoid the problem of catastrophic knowledge forgetting, which is crucial for handling long video understanding tasks. The features stored in short-term memory are dense tokens, but due to the limitations of GPU memory and computation cost, storing all the tokens dropped from short-term memory into long-term memory buffer in sequence is infeasible. Besides, we observe significant temporal redundancy in videos, where activities span multiple frames with minimal visual changes. Additionally, only a small fraction of the entire long video content is relevant to the given question in practice. To this end, in our updated version, MovieChat+, we propose to merge adjacent frames based on their relevance to specific questions, thereby streamlining video feature representation and enhancing encoding efficiency. This method converts dense tokens into sparse memories centered on pertinent questions, which are stored in long-term memory.

To be specific, as shown in Algorithm 1, we first utilize a pre-trained text encoder $\mathcal{T}(\cdot)$ to encode the specific question

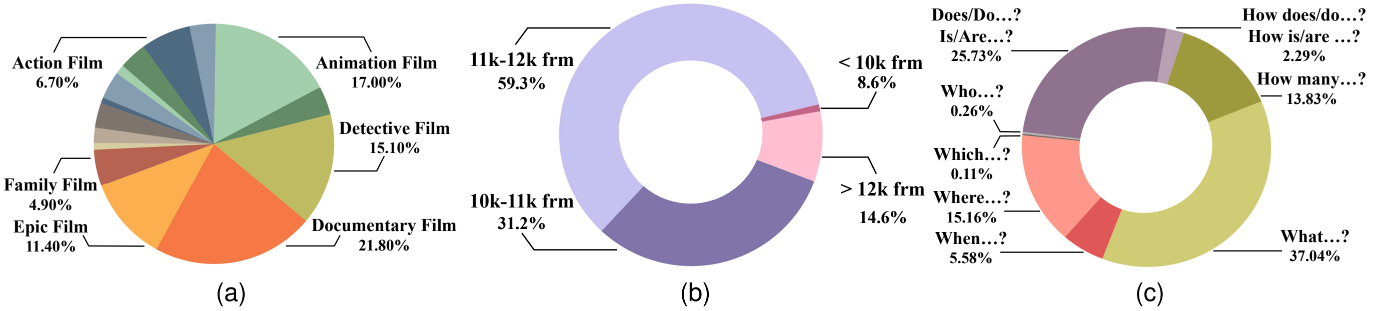


Fig. 5. Video-text statistics in MovieChat-1K. It encompasses a diverse set of categories, gathered from multiple question types and containing a diverse distribution of clip durations. "frm" represents the number of video frames.

$\mathbf{Q}$ to the same embedding space as the visual features, which can be formulated as:

$$\mathbf{q} = \mathcal{T}(\mathbf{Q}). \tag{3}$$

We then calculate the average cosine similarity s^q between each frame feature $\mathbf{x}_i$ within the short-term memory and the encoded question $\mathbf{q}$, which can be formulated as:

$$s^q = \frac{1}{K}\sum_{i=1}^{K}\left[\cos(\mathbf{x}_i, \mathbf{q})\right]. \tag{4}$$

When we watch a video with a question in mind, we tend to skip over largely irrelevant segments. This is the motivation for using the calculated similarity between questions and visual features. If the visual features in short-term memory are highly related to the questions, we merge fewer into long-term memory; otherwise, we merge more during consolidation. We consider the target merging threshold M, *i.e.*, the number of consolidated frames after merging using the similarity calculated above.

We compare the average similarity s^q to the threshold σ to assess its relevance to the question. The target merging threshold M is formulated as follows,

$$M = \begin{cases} M_0 & \text{if } s^q > \sigma \\ \alpha M_0 & \text{otherwise} \end{cases} \tag{5}$$

For segments with high relevance, we set a base merging threshold $M = M_0$. For segments with low relevance, we set a compression coefficient α, reducing the target merging threshold to $M = \alpha M_0$, thus compressing fixed-length segments into fewer ones.

Following the methodology outlined in ToMe [13], we then periodically perform memory consolidation by merging the most similar tokens in the adjacent frames. Here, we calculate the average cosine similarity s^f among N embedded tokens, as the tokens can effectively summarize the information of each frame:

$$s^f = \frac{1}{N}\sum_{j=1}^{N}\left[\cos(\mathbf{x}_i^j, \mathbf{x}_{i+1}^j)\right]. \tag{6}$$

Our goal is to keep M frames after every merge operation, which also embeds rich information stored in the long-term memory. M is the hyper-parameter to control the trade-offs between performance and efficiency. Therefore, we greedily merge each set of adjacent frames with the highest similarity based on hierarchical clustering via weighted averaging, as shown in Algorithm 1. The merge operation is iteratively conducted until the token count reaches the predefined value set M for each consolidation operation, resulting in the output video feature $\mathbf{v}' \in \mathbb{Z}^{M\times 3\times H\times W}$. Consequently, the dense tokens of the entire video are compressed to varying degrees based on their similarity to the question and are stored within the long-term memory as shown in Fig. 4. The above algorithm is parameter-free and can be easily plugged into a frame-based video encoder. Although the frame similarity calculation brings additional computing overhead, it is negligible compared to the efficiency gained by reducing stored frames.

Extend Positional Encoding. For long-term memory, the number of tokens exceeds the maximum length of the positional encoding from the pre-trained model. Thus, our model utilizes the positional encoding mechanism following BERT [35], which results in a portion exceeding the length threshold n without available positional encoding. In order to handle long enough memory, we adopt the hierarchically decomposed positional encoding method proposed by Su [66], which allows us to extend the absolute positional encoding of length from n to n^2.

3.5 Inference

Previous methods always use the representation of the whole video to conduct understanding and question-answering. While this method provides a broad overview, it often struggles with accurately localizing specific moments or details in long videos. To this end, we propose two inference modes, global and breakpoint, for long video understanding tasks as follows.

Global Mode. Global mode is defined as the understanding and question-answering for the whole video. Under this mode, the focus is on capturing the details of the full duration of the video. Therefore, we only use long-term memory $\mathcal{L}$ as the video representation $\mathbf{V}$.

Breakpoint Mode. Breakpoint mode is defined as understanding specific moments in a video. Since events possess continuity, we need to consider not only the information directly related to the moments stored in short-term memory $\mathcal{S}$ but also the information indirectly related stored in long-term memory $\mathcal{L}$. Therefore, we hypothesize that when querying the movie at a specific moment t, the video representation $\mathbf{V}$ should be the aggregation of $\mathcal{L}$, $\mathcal{S}$, and the current video frame feature $\mathbf{x}_t$. We observe that straightforward concatenation of these elements delivers outstanding

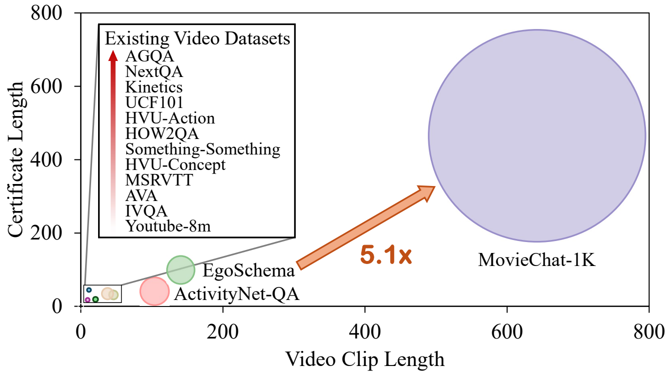


Fig. 6. Certificate length across video datasets.

TABLE 1
Semantic Categorization of MovieChat-1K Questions

Type	Rate	Frame Type	Text Rate	Type	Text Rate
Location	21.2%	Scene	11.2%	Existence	5.0%
Time	17.2%	Object	10.7%	Weather	3.6%
Number	14.6%	Action	7.8%	Others	1.5%

results, and we defer the investigation of alternative aggregation methods to future research.

Subsequently, the video representation goes through a Q-former and a linear projection layer before being fed into the LLM $\mathcal{O}$, which can be formulated as:

$$\mathbf{A}_t = \mathcal{O}(\mathbf{Q}, \mathcal{P}(\{\mathcal{L}, \mathcal{S}, \mathbf{x}_t\})), \tag{7}$$

where $\mathcal{P}$ is the projection from visual space to text space, $\mathbf{A}_t$ represents the answer or instruction of the breakpoint, and $\mathbf{Q}$ is employed to denote the question, respectively.

4 A New Benchmark: MovieChat-1K

Previous works on building long video understanding benchmarks either focus on non-question-answering tasks (*e.g.* , language grounding [64], generic event boundary detection [63], user engagement and movie metadata prediction [78], *etc.*) or lack long-form understanding evaluation [32]. To better evaluate the performance of MovieChat, we collect a new benchmark for long video understanding tasks, MovieChat-1K, which contains 1K high-quality video clips sourced from various movies and TV series with 14K manual annotations. In our updated version, MovieChat+, we expand by an additional 2K temporal grounding labels.

Video Source. As shown in Fig. 5a, we collect videos from 15 popular categories with varying distribution, including documentary film, detective film, animation film, etc. Among these, each video comprises multiple alternating scenes, contributing to a diverse and dynamic visual narrative within the context of the collection. We further illustrate our improved content-based categorization and analysis of MovieChat-1K questions in Tab. 1. The visual representation in Fig. 5b demonstrates the clip duration distribution of MovieChat-1K. Over 90% of the videos exhibit a duration ranging from 10K to 12K frames, while 14.6% of videos extend beyond 12K frames. Only 8.6% of videos have a duration of less than 10k frames. To demonstrate that MovieChat-1K is indeed a long-form dataset, we employ the same method as proposed by EgoSchema [49] to calculate the certificate lengths. As depicted in Fig. 6, our approach results in a certificate length that is 5.1 times longer than that of EgoSchema [49]. Specifically, the annotated captions and questions are used as temporal tags, and the corresponding clip lengths are calculated manually.

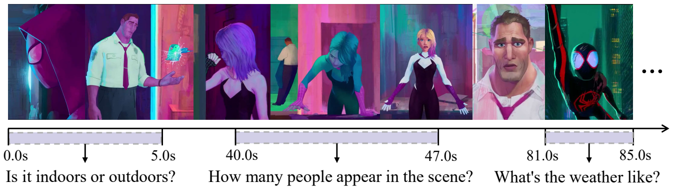

...
0.0s
5.0s
Is it indoors or outdoors?
40.0s
47.0s
How many people appear in the scene?
81.0s
85.0s
What's the weather like?

Fig. 7. Examples of temporal labels in MovieChat-1K.

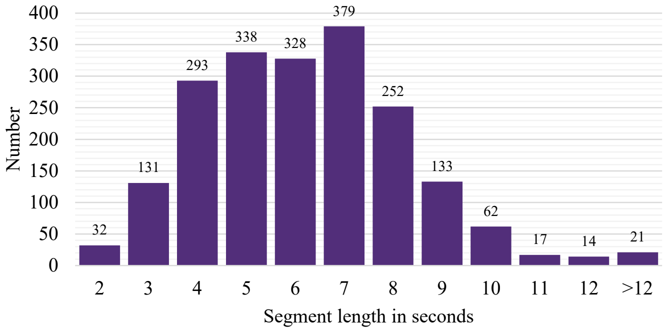


Fig. 8. Distribution of temporal segments.

Temporal Label Collection. Following [81], we augment MovieChat-1K with temporal labels. Most VideoQA datasets [34], [83] are unsuitable for exploring how to handle irrelevant redundant frames, as they are composed of short video clips (no more than 15 seconds) that have been pre-trimmed to focus solely on the pertinent content. We apply temporal labels exclusively to questions categorized under breakpoint mode. This is because questions in global mode mostly pertain to global video content (*e.g.* , ”Where does the video take place?”). Furthermore, the answers to these global mode questions are often discernible across extensive segments of the video, such as “Is there more than five different characters appearing?”. For each question-answer pair in breakpoint mode, we annotate the start and end times of the relevant segments as shown in Fig. 7.

We restrict the labeling of temporal annotations in MovieChat-1K to the validation and test sets, under the premise that these labels are instrumental in assessing the ability of models to identify question-relevant video segments, rather than for training purposes. As a result, 2K question-answer pairs drawn from 200 videos are annotated with temporal labels. Fig. 8 demonstrates that most of the segments last for less than 12 seconds, with an average duration of 6.3 seconds, which is extremely short compared to the video length (approximately 700 seconds).

Annotations Analysis. For each video, we manually set and provide 1 dense caption for the whole video, 3 question-answering pairs for global mode, and 10 question-answering pairs with timestamps for breakpoint mode. Fig. 5c illustrates the distribution of question types in

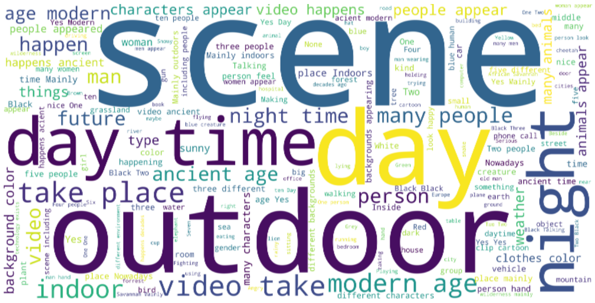


Fig. 9. Word cloud of the answer set in MovieChat-1K.

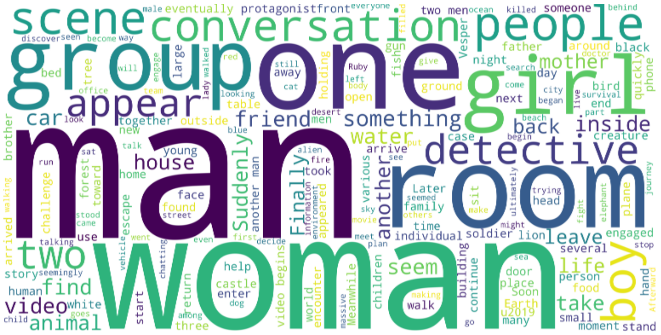


Fig. 11. Word cloud of the caption set in MovieChat-1K.

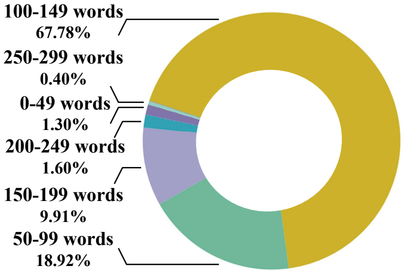


Fig. 10. Distribution of caption length.

TABLE 2
Quantitative evaluation for short video question answering with GPT-3.5 [51]. MovieChat achieves comparable performance even it is not specifically designed for for short video question-answering tasks. The best result is highlighted in bold, and the second best is underlined. Acc. and Sco. stand for accuracy and score respectively.

Method	MSVD-QA		MSRVTT-QA		ActivityNet-QA		NExT-QA	
	Acc.	Sco.	Acc.	Sco.	Acc.	Sco.	Acc.	Sco.
FrozenBiLM [86]	2.2	–	16.8	–	24.7	–	–	–
Video Chat [41]	56.3	2.8	45.0	2.5	26.5	2.2	**56.6**	**3.2**
LLaMA Adapter [93]	54.9	3.1	43.8	2.7	34.2	2.7	–	–
Video LLaMA [92]	51.6	2.5	29.6	1.8	12.4	1.1	–	–
Video-ChatGPT [48]	64.9	3.3	49.3	**2.8**	35.2	2.7	54.6	**3.2**
MovieChat	75.2	3.8	52.7	2.6	45.7	**3.4**	49.9	2.7
MovieChat+	**76.5**	**3.9**	**53.9**	2.7	**48.1**	**3.4**	54.8	3.0

MovieChat-1K. Note that MovieChat-1K is specifically designed for long video comprehension tasks. The majority of questions are open-ended, with only a quarter classified as multiple-choice questions, marked by initiators such as ‘Do,’ ‘Does,’ ‘Is,’ or ‘Are.’ As illustrated in Fig. 9, we also compute the word distributions of the question-answer pairs, which includes common objects (people, clothes, etc.), time (day, night, etc.), scenes (indoor, outdoor, etc.), and so on.

To facilitate a detailed understanding of long videos, we provide a dense caption for each video. As shown in Fig. 10, MovieChat-1K exhibits diverse caption lengths in the segmented clip level. Approximately two-thirds of the clips have captions with 100-149 words, while one-fifth of the clip captions have fewer than 100 words. About 11% of clips have long captions with more than 150 words.

To analyze the word distribution of our generated captions, we compute their distributions. The resulting word distribution is presented in Fig. 11, which includes common objects (man, woman, people, girl, etc.), attributes (detective, various, small, white, etc.), locations (inside, behind, south, next, etc.), scenes (room, house, building, office, etc.), actions/events (talk, enter, leave, take, etc.), and more.

In terms of actions, MovieChat-1K captions contain nearly the same number of verbs as with the WebVid10M dataset [12]. To evaluate this, we use the NLTK toolkit to analyze the number of verbs in captions, focusing on extracting and tagging all unique verbs. We find a total of 109,485 verbs in the WebVid10M caption dataset, while the MovieChat-1K captions contain 102,988 unique instances of verbs. While these counts may not be entirely accurate due to our simple counting method, we believe they provide a rough indication of the actions of the two datasets.

5 Experiments

We conduct quantitative and qualitative evaluations of our complete MovieChat+ compared to previous methods and the original MovieChat. Additionally, we perform ablation studies to investigate MovieChat+.

5.1 Quantitative Evaluation

5.1.1 Short Video Question-answering

We use several widely used open-ended datasets: MSVD-QA [83], MSRVTT-QA [85], ActivityNet-QA [89], and NExT-QA [80] for short video question-answering tasks. The evaluation process is under the assistance of LLM with the default hyper-parameter settings. The accuracy and relative scores on a scale of 0 to 5 are reported. Compared to previous methods [41], [48], [86], [92], [93], MovieChat achieves comparable performance even it is not specifically designed for short video question-answering tasks, as shown in Tab. 2.

We also report the results of our zero-shot evaluation on the test split of the NExT-QA [80] benchmark in Tab. 3. NExT-QA divides its questions into three categories: Causal (C), Temporal (T), and Description (D). Compared to prior work, our approach achieves higher accuracy across all aspects, demonstrating its effectiveness at understanding temporal context with the question-aware consolidation.

Following [48], we employ GPT-assisted evaluation to conduct a more comprehensive comparison of the text generation performance between our appraoch and previous

TABLE 3
Zero-shot evaluation on NExT-QA [80] test split. We observe that our approach performs better than other approaches across most of the different video understanding tasks. Acc stans for the accuracy.The best result is highlighted in bold, and the second best is underlined.

Method	Acc$_C$	Acc$_T$	Acc$_D$	Acc$_{AVG}$
Video LLaMA (finetuned)	27.43	32.14	32.38	29.71
VideoLLama	31.32	35.49	42.64	34.47
MovieChat	31.12	35.80	42.49	34.43
MovieChat+	**32.45**	**36.03**	**43.58**	**35.21**

TABLE 4
Quantitative evaluation for short video generation performance with GPT-3.5 [51]. CI stands for correctness of information, DO stands for detail orientation, CU stands for contextual understanding, TU stands for temporal understanding, and CO stands for consistency. The best result is highlighted in bold, and the second best is underlined.

Method	CI	DO	CU	TU	CO
Video Chat [41]	2.23	2.50	2.53	1.94	2.24
LLaMA Adapter [93]	2.03	2.32	2.30	1.98	2.15
Video LLaMA [92]	1.96	2.18	2.16	1.82	1.79
Video-ChatGPT [48]	2.40	2.52	2.62	1.98	2.37
MovieChat	2.76	2.93	3.01	2.24	2.42
MovieChat+	**2.87**	**2.95**	**3.10**	**2.25**	**2.50**

methods [41], [48], [92], [93] on processed ActivityNet-QA [89]. The evaluation pipeline covers crucial metrics (including *Correctness of Information*, *Detailed Orientation*, *Contextual Understanding*, *Temporal Understanding* and *Consistency*) and assigns relative scores to the generated predictions on a scale of 0-5. We present the results of the generation performance evaluation in Tab. 4. The results reveal its competitive performance across all key aspects compared to previous methods. It should be noted that in comparison with MovieChat, MovieChat+ does not exhibit substantial enhancements in both question-answering accuracy and generative performance when evaluated on short video datasets. This is because the content of short videos is often closely related to the questions, which is an extreme case for our proposed frame filtering strategy. Similar to the original MovieChat, all video frames are merged with nearly equal consideration.

TABLE 5
Zero-shot video question answering on Seed-Bench [36]. The “Frames” column lists the frames input into the LLM decoder. The best result is highlighted in bold, and the second best is underlined.

Method	# Frames	Procedure Understanding	Action Recognition
Video Chat [41]	32	27.27	34.89
Video LLaMA [92]	32	25.42	35.52
Video-ChatGPT [48]	32	21.14	27.59
MovieChat	32	26.76	34.37
	256	29.82	40.11
MovieChat+	32	27.35	36.33
	256	**31.04**	**42.45**

TABLE 6
Quantitative evaluation for long video question answering on MovieChat-1K test set in global mode with the average of GPT-3.5 [51], Claude [4] and human bling rating. ”# Frames” indicates the count of video frames read by the models. The best result is highlighted in bold, and the second best is underlined. Acc. and Sco. stand for accuracy and score respectively.

Method	Text Decoder	# Frames	Global Mode		Breakpoint Mode	
			Acc.	Sco.	Acc.	Sco.
GIT [74]	non-LLM based	6	28.8	1.83	29.2	1.98
mPLUG-2 [84]	non-LLM based	8	31.7	2.13	30.8	1.83
Video Chat [41]	LLM based	32	57.8	3.00	46.1	2.29
Video LLaMA [92]	LLM based	32	51.7	2.67	39.1	2.04
Video-ChatGPT [48]	LLM based	100	47.6	2.55	48.0	2.45
MovieChat	LLM based	2048	62.3	3.23	48.3	2.57
MovieChat+	LLM based	2048	**71.2**	**3.51**	**49.6**	**2.62**

We further evaluate our approach on the task of action recognition on Seed-Bench [36] to study the effect of MovieChat for short-term temporal understanding tasks. In contrast to the longer setting in the procedure understanding task, the videos in this task generally have duration of around 10 seconds. As shown in Fig. 5, we compile the number of frames fed into the LLM decoder for different models, along with the corresponding performance of procedure understanding and action recognition. As the input frames increasing, our question-aware approach yields greater benefits in both the procedure understanding task and the action recognition task. These results suggest that the question-aware merge strategy, which filters more related context for reasoning about spatial-temporal relationships between video segments, may be crucial for fine-grained action understanding. However, when feeding the same number of frames into the LLM decoder, MovieChat shows minimal improvement in procedure understanding. We speculate that when dealing with a limited number of sampling frames, the inclusion of compressed frames containing redundant information could potentially hinder the model’s ability to comprehend procedural sequences.

5.1.2 *Long Video Question-answering*

We evaluate the long video question-answering performance of MovieChat with our proposed MovieChat-1K. We split 1,000 videos into training set (800), test set (100), validation set (100) and only use test set for final performance evaluation. We select two non-LLM based video understanding models (*e.g.* GIT [74], and mPLUG-2 [84]) and three recent LLM-based video understanding models (*e.g.* Video Chat [41], Video LLaMA [92], and Video-ChatGPT [48]) as the baselines. Yet, none of those methods can support such long video (>10K frames). Therefore, to accommodate their length limitations in global questions, we uniformly sample from the original video up to the maximum frame count which can be officially supported by each individual model. For breakpoint questions, we extend half of the maximum frame count before and after the breakpoint (placing the breakpoint at the center frame).

To enhance the robustness of the results, we simultaneously employ GPT-3.5 [51] and Claude [4] as LLM assistants, with the additional support of human blind rating. We observe a discrepancy between the accuracy and relative score generated by the previously LLM-assisted evaluation method [48] for video question-answering tasks. However, merely adjusting the prompt for the LLM cannot effectively

TABLE 7
Quantitative evaluation for long video generation performance in global mode with the average of GPT-3.5 [51], Claude [4] and human blind rating. CI stands for correctness of information, DO stands for detail orientation, CU stands for contextual understanding, TU stands for temporal understanding, and CO stands for consistency. The best result is in bold, and the second best is underlined.

Method	Text Decoder	CI	DO	CU	TU	CO
GIT [74]	non-LLM based	1.65	1.74	2.00	1.88	2.16
mPLUG-2 [84]	non-LLM based	1.82	1.93	2.24	1.92	2.35
Video Chat [41]	LLM based	3.04	2.75	3.09	3.00	3.21
Video LLaMA [92]	LLM based	2.75	2.24	2.83	2.62	2.97
Video-ChatGPT [48]	LLM based	2.37	2.30	2.58	2.49	2.69
MovieChat	LLM based	3.11	2.93	3.24	3.17	3.25
MovieChat+	LLM based	**3.28**	**3.04**	**3.47**	**3.22**	**3.41**

TABLE 8
Quantitative evaluation for long video generation performance in breakpoint mode with the average of GPT-3.5 [51], Claude [4] and human blind rating. CI stands for correctness of information, DO stands for detail orientation, CU stands for contextual understanding, TU stands for temporal understanding, and CO stands for consistency. The best result is highlighted in bold, and the second best is underlined.

Method	Text Decoder	CI	DO	CU	TU	CO
GIT [74]	non-LLM based	1.44	1.48	1.96	1.77	2.01
mPLUG-2 [84]	non-LLM based	1.53	1.62	1.91	1.56	2.14
Video Chat [41]	LLM based	2.42	2.51	2.81	2.10	2.78
Video LLaMA [92]	LLM based	2.04	2.29	2.63	2.00	2.87
Video-ChatGPT [48]	LLM based	2.62	**2.65**	2.86	2.32	2.96
MovieChat	LLM based	2.64	2.60	**2.87**	2.49	**3.08**
MovieChat+	non-LLM based	**2.65**	2.62	**2.87**	**2.51**	3.07

TABLE 9
Quantitative evaluation for long video different types question answering in global mode. The best result is highlighted in bold, and the second best is underlined. Acc. and Sco. stand for accuracy and score respectively.

Method	Total		Multi-choice		Open-ended	
	Acc.	Sco.	Acc.	Sco.	Acc.	Sco.
Video Chat [41]	61.0	3.34	74.8	3.83	56.4	3.02
Video LLaMA [92]	51.4	3.10	78.3	3.58	38.8	2.67
Video-ChatGPT [48]	44.2	2.71	52.5	3.16	37.7	2.54
MovieChat	62.3	3.81	80.9	4.02	57.5	3.74
MovieChat+	**71.2**	**3.51**	**81.4**	**4.03**	**60.1**	**3.79**

TABLE 10
Quantitative evaluation for long video different types question answering in breakpoint mode. The best result is highlighted in bold, and the second best is underlined. Acc. and Sco. stand for accuracy and score respectively.

Method	Total		Multi-choice		Open-ended	
	Acc	Sco.	Acc	Sco.	Acc	Sco.
Video Chat [41]	48.3	2.43	62.4	3.46	44.5	2.19
Video LLaMA [92]	38.2	2.33	57.3	2.39	33.1	2.31
Video-ChatGPT [48]	49.8	2.71	58.3	3.05	47.5	2.37
MovieChat	48.3	2.57	62.4	3.65	46.7	2.70
MovieChat+	**49.6**	**2.62**	**64.0**	**3.79**	**47.7**	**2.73**

address this issue. Therefore, after obtaining the accuracy and score from the LLM-assisted evaluation method, we implement manual filtering to remove results with inconsistent values, thus improving the reliability of our outcomes.

As shown in Tab. 6, compared to previous methods, MovieChat reads more video frames. In both global mode and breakpoint mode, our method maintains a performance gain in terms of the average accuracy and score provided by LLM assistants and human blind rating. Compared with MovieChat, our method significantly improves accuracy in the global mode, which fully demonstrates the effectiveness of our question-aware consolidation strategy.

We further compare the quality of answers generated by MovieChat and previous methods [41], [48], [74], [84], [92] in long video question-answering on MovieChat-1K. As shown in Tab. 7 and Tab. 8, with the average score provided by GPT-3.5 [51], Claude [4] and human bling rating, our complete method, MovieChat+, continues to generate higher-quality answers even as the video contents become more extensive, significantly outperforming the initial and simpler version of MovieChat.

MovieChat-1K contains question-answer pairs of varies types. To better assess the performance of our approach, we conduct evaluations on the long video question answering task using various types of questions. We roughly categorize the question types into multiple-choice questions and open-ended questions. With the average results of GPT-3.5 [51], Claude [4] and human blind rating, Tab. 9 and Tab. 10 respectively present the accuracy and scores of MovieChat and the baseline across different question categories in both global mode and breakpoint mode. In various research conditions, our approach consistently outperforms the baselines in both open-ended and true-false questions.

5.1.3 *Question-answering on Other Long Video Dataset*

We further report zero-shot question-answering results for MovieChat on another commonly used long-form video dataset, EgoSchema [49] in Tab. 11. EgoSchema [49] is a diagnostic benchmark for evaluating long video understanding capabilities of advancing systems, featuring over 5000 human-curated multiple-choice question-answer pairs based on more than 250 hours of real-world video data. Prior works [3], [53], [98] have demonstrated that the sequence in which options are presented can significantly influence the outcomes of tasks involving multiple choices. To mitigate this effect, we provide MovieChat with questions in EgoSchema [49]exclusively. Subsequently, we employ LangChain [33] to assess the similarity between the responses of MovieChat and the provided options. We then select the option that most closely aligns with our anticipated answer as our decision. MovieChat+ produces significantly superior result than its initial version and other leading non-LLM based and LLM-based methods.

5.2 Ablation Study

5.2.1 *Short-term and Long-term Memory Buffers*

As our approach incorporates a memory mechanism including short-term memory and long-term memory, it is imperative to evaluate how the proposed memory mechanism influences the performance. Tab. 12 and Tab. 13 provide the memory-dependent performance of our approach for long video question-answering and generative tasks with the average results of GPT-3.5 [51], Claude [4], and human blind rating. MovieChat with the memory mechanism

TABLE 11
Zero-shot QA Evaluation on EgoSchema. Acc. stands for the accuracy. The best result is highlighted in bold, and the second best is underlined.

Model	Text Decoder	Acc.
Choosing the correct answer uniformly at random		20.0%
FrozenBiLM [86]	non-LLM based	26.9
Video Chat [41]	LLM based	47.5
MovieChat	LLM based	53.5
MovieChat+	LLM based	<u>56.4</u>
Human Performance		**76.2%**

TABLE 12
Ablation study on how memory mechanism (MM) affects the long video question answering. The best result is in bold.

Method	Global Mode		Breakpoint Mode	
	Accuracy	Score	Accuracy	Score
w/o MM	51.4	3.10	38.2	2.31
base	**67.8**	**3.81**	**50.4**	**2.96**

TABLE 13
Ablation study on how memory mechanism (MM) affects the long video generative performance. CI stands for correctness of information, DO stands for detail orientation, CU stands for contextual understanding, TU stands for temporal understanding, and CO stands for consistency. The best result is in bold.

Method	Global Mode					Breakpoint Mode				
	CI	DO	CU	TU	CO	CI	DO	CU	TU	CO
w/o MM	3.30	2.53	3.28	2.77	3.42	2.42	2.85	2.87	2.00	2.87
base	**3.32**	**3.28**	**3.40**	**2.97**	**3.48**	**2.97**	**3.24**	**3.31**	**2.70**	**3.45**

TABLE 14
Ablation on memory modules. We show CIDEr on ActivityNet [89] under different input frames T. The second column shows the number of input tokens to the LLM decoder. $N_f = 257$ is the number of tokens per-frame. $K = 514$ is the number of memory tokens. The best result is highlighted in bold, and the second best is underlined.

Method	# Tokens	$T = 16$	$T = 32$	$T = 64$	$T = 128$
No Memory	$T \times N_f$	**29.8**	-	-	-
Spatial Pooling	T	27.6	27.3	<u>27.9</u>	<u>27.4</u>
Temporal Pooling	N_f	<u>29.3</u>	<u>28.0</u>	26.8	25.2
EMA [94]	N_f	28.2	26.3	22.0	16.3
Ours	K	<u>29.3</u>	**29.2**	**28.9**	**29.3**

significantly outperforms the memory-independent variant, which signifies the importance of memory mechanisms.

We further consider the following approaches to evaluate our memory consolidation strategy:

No memory. Due to memory constraints, we uniformly sample 16 frames from all frames, concatenate all visual tokens, and feed them into the decoder.

Spatial- or temporal-pooling. We pool the visual features, along either the spatial or temporal dimensions to reduce the number of tokens fed to the LLM decoder.

EMA. Following [99], we use an exponential moving average of frame features at each time step.

Tab. 14 compares the results of the different memory modules. For $T = 16$, where we can feed all the tokens from the vision backbone into the decoder, "No Memory" performs the best because it uses the most tokens. However, it is impractical to use "No Memory" for $T>16$ due to its computational cost. With more frames, naively pooling along the spatial- or temporal-dimensions actually performs worse. This is likely because we are averaging out information over longer temporal durations, and thus losing the important details required for more detailed localization or captioning. Our method, on the other hand, leverages more frames to improve performance by retaining diverse features within the memory.

5.2.2 Large Language Models Ablations

Most previous video understanding methods [41], [48], [92], [93] primarily employed LLama [70] and its variants [21] as text decoders. With the average results of GPT-3.5 [51], Claude [4] and human blind rating, Tab. 15 and Tab. 16 illustrate how the performance of MovieChat changes when using LLama [70] and LLama2 [71] as the large language model respectively.

Contrary to our initial hypothesis, the performance of MovieChat with LLama2 [71] hardly surpasses those of MovieChat with LLama [70] across various key metrics. The outcome suggests that the advancements incorporated into LLama2 [71] may not translate to significant improvements. We further investigate a specific example to analyze this phenomenon. As shown in Fig. 13, MovieChat with LLama [70] provides answers that are more aligned with the video content. Surprisingly, MovieChat with LLama2 [71] offers an approximation of the time required for each step (indicated in *italics*). While its time estimates do not precisely match the actual durations, the proportion of time provided is realistic. Even though LLama2 [71] cannot obtain specific time information when processing feature-rich video frames, the memory buffer design allows for dense sampling of video frames, enabling LLama2 [71] to estimate the proportion of time for each scene based on adjacent similar frames. Therefore, we propose that the lower evaluation metric results of MovieChat with LLama2 [71] compared to MovieChat with LLama [70] may be attributed to the question-answer pairs in the dataset.

5.2.3 Hyper-parameter Ablations

We perform a series of hyper-parameter ablations based on the MovieChat-1K dataset to better understand our approach. Fig. 12 shows the performance when ablating the length of long-term memory buffer l_{long}, the length of short-term memory buffer l_{short}, short-term initialization, question-frame similarity σ, target merging coefficient α and judging relevance basis with the average results of GPT-3.5 [51], Claude [4], and human blind rating. The performance of MovieChat degrades which shows the validity of our empirically chosen hyper-parameters.

Length of Memory Buffer. The length of different memory buffers has a combined effect on MovieChat's performance. Since the LLM-based evaluation shows a positive correlation between accuracy and score, we use accuracy to gauge performance. Fig. 12 (top left and middle) demonstrates that information obtained from the video expands with the growing length of memory buffers. However, this benefit

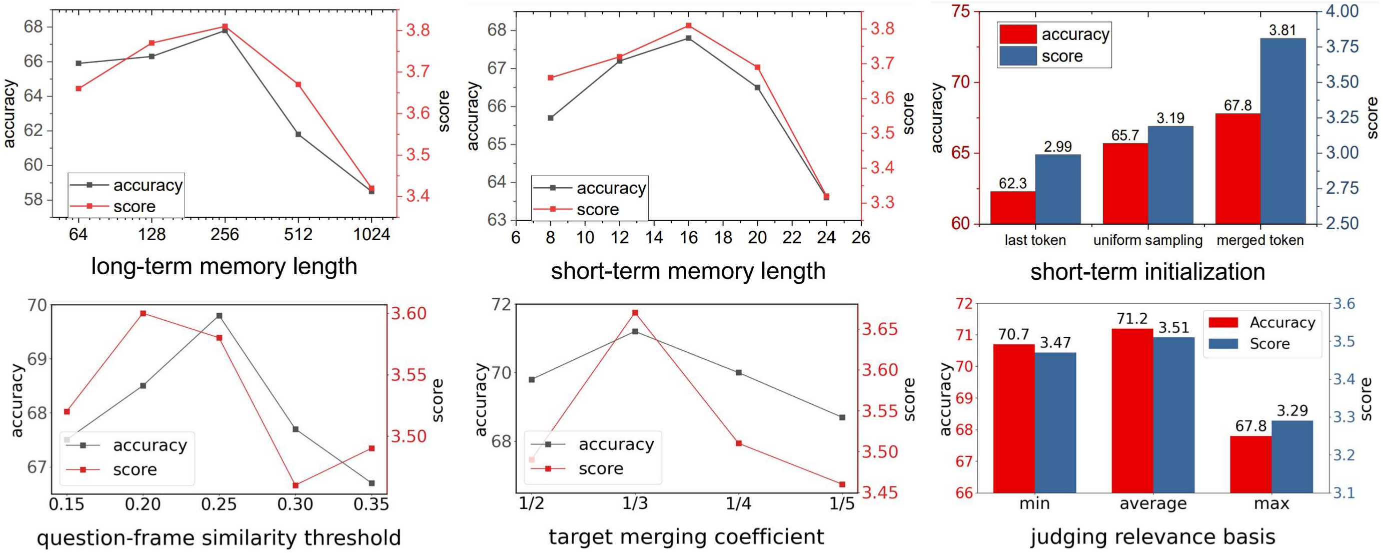


Fig. 12. Hyperparameter ablation studies on how length of memory buffer l_{long} and l_{short}, short-term initialization, question-frame similarity threshold σ, target merging coefficient α and judging relevance basis affect the performance of MovieChat+ on long video understanding.

TABLE 15
Ablation Study on how LLM affects the long video question answering. The best result is highlighted in bold.

Method	Global Mode		Breakpoint Mode	
	Accuracy	Score	Accuracy	Score
LLama [70]	**67.8**	**3.81**	**50.4**	2.96
LLama2 [71]	64.2	3.79	48.1	**2.98**

TABLE 16
Ablation Study on how the large language model affects the long video generative performance. MM stands for memory mechanism, CI stands for correctness of information, DO stands for detail orientation, CU stands for contextual understanding, TU stands for temporal understanding, and CO stands for consistency. The best result is highlighted in bold.

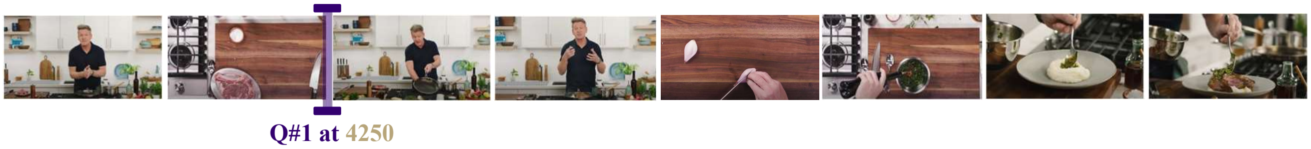
Method	Global Mode					Breakpoint Mode				
	CI	DO	CU	TU	CO	CI	DO	CU	TU	CO
LLama [70]	**3.32**	**3.28**	3.40	**2.97**	**3.48**	**2.97**	**3.24**	3.31	**2.70**	3.45
LLama2 [71]	3.27	**3.28**	**3.41**	2.95	3.45	2.96	3.12	**3.38**	2.68	**3.34**

is tempered by a more pronounced loss of fine details, a consequence of maintaining a fixed length for consolidation. Therefore, as the lengths of two memory buffers increase, the performance of our approach exhibits a trend of initially rising and then declining. Thus, we set the length of long/short-term memory to 256 and 16, respectively.

Short-term Initialization. As shown in Fig. 12 (top right), using merged tokens for short-term initialization outperforms the last few tokens and uniform sampling. When initializing the next short-term memory with the last few tokens from the previous short-term memory, it is unable to adequately represent the previous information, leading to the final merged tokens being either repetitive or lacking coherence with the previous time step. Uniform sampling faces similar issues, but it manages to capture information with representative frames from the previous time step.

Question-frame Similarity Threshold. Before compressing the short-term memory, we need to assess the relevance of the stored segments to the question by evaluating the similarity between video frames and the question, thereby deciding the degree of compression for the current short-term memory. Fig. 12 (bottom left) illustrates the outcomes of experimenting with various question-frame similarity thresholds, and the optimal performance is achieved at $\sigma = 0.25$. When the threshold σ is low, it is difficult to effectively compress and filter out distracting or redundant information. Conversely, an excessively high threshold σ might lead to the over-compression of valuable information.

Target Merging Coefficient. We further explore the target merging coefficient α of weakly to strongly related segments. For fairness, all weakly related short-term memories are compressed into 1 frame, allowing us to focus on how the performance of strongly related segments varies with different compression levels. Fig. 12 (bottom middle) shows that increasing merged frames for strongly related segments initially boosts performance but eventually leads to a decline. We initially assume that less compression of strongly related segments would significantly enhance model performance. Yet, the results hint at a more intricate link between compression intensity and performance. To incorporate more long-term memory frames into the pre-trained model, we extend positional encoding with hierarchical decomposition. However, the approach involves balancing extended input lengths with the integrity of positional representations. A direct extension may not be ideal since most training frames are shorter than those in the pre-trained model, making lower positions well-trained for absolute positions, whereas higher positions are less trained, offering only a rough estimate of relative positions. Thus, interpolating lower positions poses a greater risk of disrupting established positional embeddings compared to interpolating higher positions. When querying the same video



Question #1 **What may happen next step?**

Video Chat Based on the video, it appears that the person is cooking meat and vegetables on a stove top using different utensils and ingredients such as onions, garlic, mustard, and olive oil. **It is likely that the person will continue cooking the meat until it is fully cooked, and then serve it to their family or friends.** Additionally, they may add other ingredients such as mushrooms or potatoes to the dish. It is possible that they will use different types of cooking methods such as sautéing or grilling to ensure the meat is cooked evenly and to perfection.

Video LLaMA At frame 4250, the video shows the hand of a man cutting meat on a cutting board. It's not clear what happens next, but it is possible that the man continues cutting the meat.

Video-ChatGPT The next step in the video is to place the cooked meat on a plate and serve it. The person in the video is cutting the meat into smaller pieces and placing it on a plate. The person then takes a fork and begins to eat the meat.

MovieChat **I think the man will prepare the food in the kitchen. It seems like he is going to cook it with spices and herbs.** He has a large wooden board on the kitchen counter and a large white pot on the stove, and **he will probably use some tools to prepare the food. He might also use his hands or a spatula to cook the meat.**

MovieChat+ **It seems the gentleman is planning to cook in the kitchen, likely to use spices and herbs to flavor the food.** On the counter, there is a big wooden chopping board, and on the stove, there is a large white cooking pot. **He might need different tools for the cooking the meat, like knives or spoons. It looks like he is all set to make a tasty meal.**

Question #2 **Which step took the longest to complete this dish?**

Video Chat Based on the video, it appears that the longest step in making the dish is preparing the pan for cooking the steak. Specifically, the man takes the pan out of the oven and starts cleaning it by wiping it with a rag, then rubbing it with olive oil, and finally heating the pan to high heat. It is also mentioned that the man has to wait for the pan to heat up before putting the steak in the pan.

Video LLaMA The dish that this video portrays took several minutes to complete. However, **the most time-consuming task seems to have been preparing the ingredients on the white kitchen counter.** The man in the video is seen chopping vegetables and using a sharp knife to prepare a meal. Finally, he will sit down to enjoy the meal, savoring each bite and reflecting on the process that led to this culinary creation.

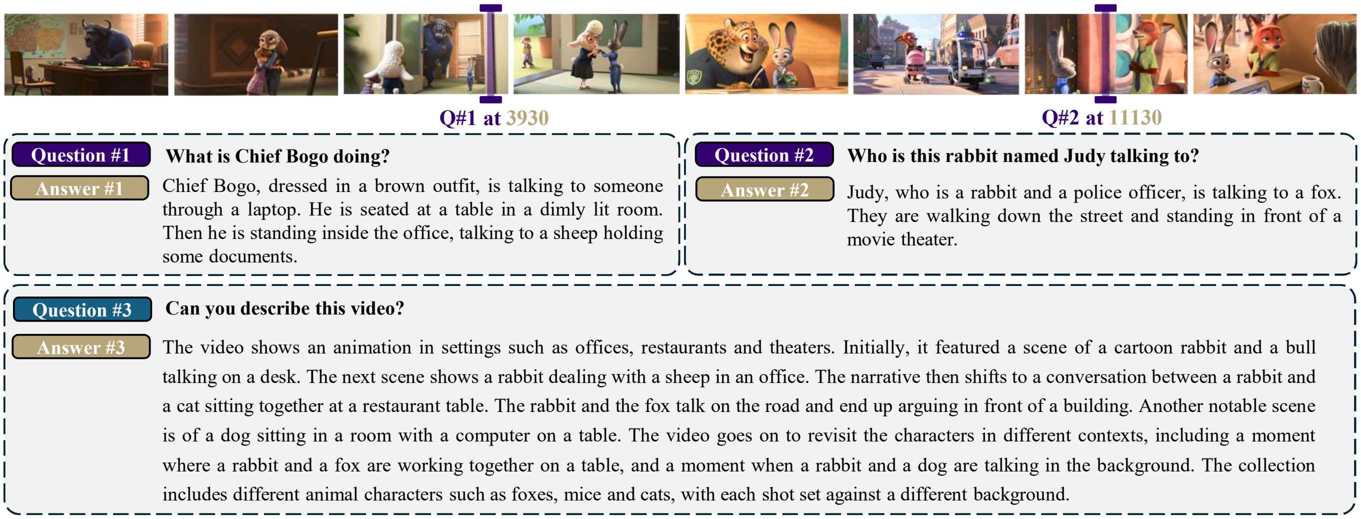
Video-ChatGPT The longest step in preparing the dish was the time it took to cook the meat. The man in the video cooks the meat on a stove for a long time, which is a crucial step in the process. The video shows the man cooking the meat on a stove for a considerable amount of time, which indicates that it took a considerable amount of effort and patience to achieve the desired result.

MovieChat **The longest step to complete this dish would be the preparation of the food on the cutting board.** It requires chopping the meat and adding other ingredients like broccoli and carrots, and mixing them together before putting them in the pan. The chef had to carefully chop and arrange the ingredients before cooking the dish.

MovieChat+ **The longest step to make this meal involves prepping everything on the chopping board. This step includes cutting up the meat, then adding veggies like broccoli and carrots.** Everything has to be chopped up and mixed together before it all goes into the pan.

MovieChat+ (LLaMA2) The step that took the longest to complete this dish was the cooking the meat on the stovetop. *It took around 30 minutes to complete this step, followed by 20 minutes to prepare the ingredients and cook the pasta, and 10 minutes to cook the meat and vegetables.*

Fig. 13. Question and answer about a clip from *YouTube*, which is a tutorial on how to cook steak. The entire instructional process begins with marinating the steak, followed by pan-searing it, preparing side dishes, and ultimately plating the meal. The correct answer is highlighted in bold, the wrong answer or the hallucinating part is underlined, and the answer with an approximation of the time required is indicated in italics.



Question #1 What is Chief Bogo doing?

Answer #1 Chief Bogo, dressed in a brown outfit, is talking to someone through a laptop. He is seated at a table in a dimly lit room. Then he is standing inside the office, talking to a sheep holding some documents.

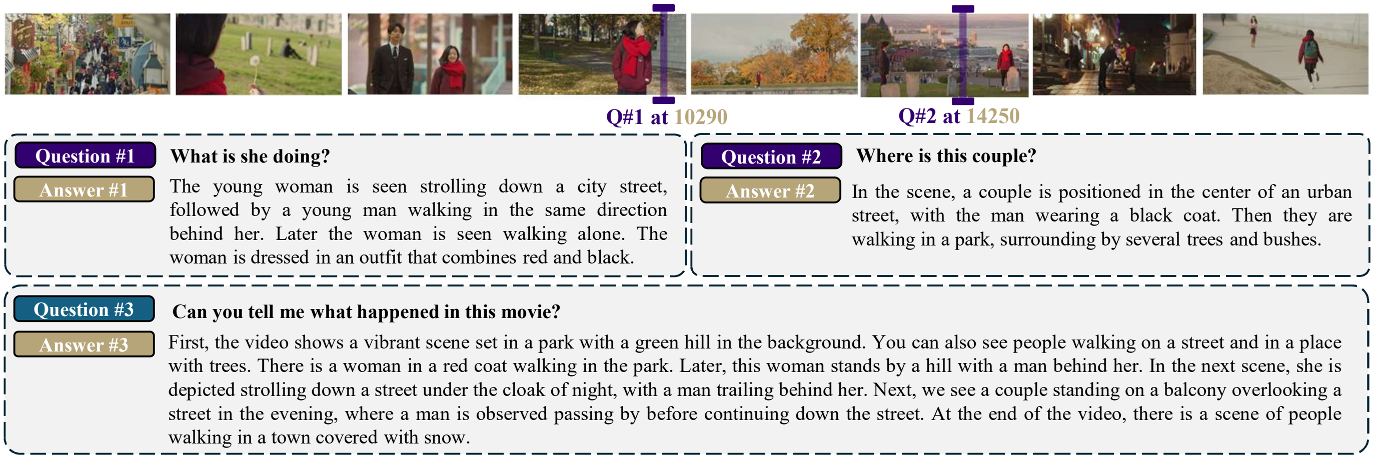
Question #2 Who is this rabbit named Judy talking to?

Answer #2 Judy, who is a rabbit and a police officer, is talking to a fox. They are walking down the street and standing in front of a movie theater.

Question #3 Can you describe this video?

Answer #3 The video shows an animation in settings such as offices, restaurants and theaters. Initially, it featured a scene of a cartoon rabbit and a bull talking on a desk. The next scene shows a rabbit dealing with a sheep in an office. The narrative then shifts to a conversation between a rabbit and a cat sitting together at a restaurant table. The rabbit and the fox talk on the road and end up arguing in front of a building. Another notable scene is of a dog sitting in a room with a computer on a table. The video goes on to revisit the characters in different contexts, including a moment where a rabbit and a fox are working together on a table, and a moment when a rabbit and a dog are talking in the background. The collection includes different animal characters such as foxes, mice and cats, with each shot set against a different background.

Fig. 14. Question and answer about clips from *Zootopia*, a cartoon, which tells the story of a determined police officer rabbit named Judy who pairs up with a cunning fox to uncover a conspiracy about missing animals and develop an unexpected friendship.



Question #1 What is she doing?

Answer #1 The young woman is seen strolling down a city street, followed by a young man walking in the same direction behind her. Later the woman is seen walking alone. The woman is dressed in an outfit that combines red and black.

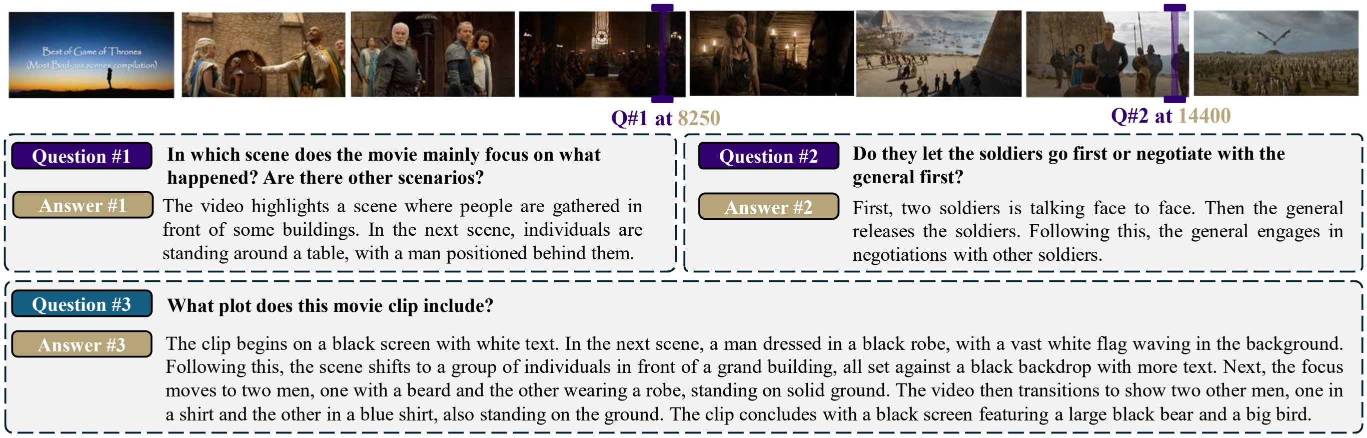
Question #2 Where is this couple?

Answer #2 In the scene, a couple is positioned in the center of an urban street, with the man wearing a black coat. Then they are walking in a park, surrounding by several trees and bushes.

Question #3 Can you tell me what happened in this movie?

Answer #3 First, the video shows a vibrant scene set in a park with a green hill in the background. You can also see people walking on a street and in a place with trees. There is a woman in a red coat walking in the park. Later, this woman stands by a hill with a man behind her. In the next scene, she is depicted strolling down a street under the cloak of night, with a man trailing behind her. Next, we see a couple standing on a balcony overlooking a street in the evening, where a man is observed passing by before continuing down the street. At the end of the video, there is a scene of people walking in a town covered with snow.

Fig. 15. Question and answer about clips from *Goblin*, which tells the story of Kim Shin, an immortal "goblin" who needs to find a human bride to end his endless life but instead meets Ji Eun-tak, a girl fated to die who claims to be the "goblin's bride," leading to a romantic tale unfolding between them.



Question #1 In which scene does the movie mainly focus on what happened? Are there other scenarios?

Answer #1 The video highlights a scene where people are gathered in front of some buildings. In the next scene, individuals are standing around a table, with a man positioned behind them.

Question #2 Do they let the soldiers go first or negotiate with the general first?

Answer #2 First, two soldiers is talking face to face. Then the general releases the soldiers. Following this, the general engages in negotiations with other soldiers.

Question #3 What plot does this movie clip include?

Answer #3 The clip begins on a black screen with white text. In the next scene, a man dressed in a black robe, with a vast white flag waving in the background. Following this, the scene shifts to a group of individuals in front of a grand building, all set against a black backdrop with more text. Next, the focus moves to two men, one with a beard and the other wearing a robe, standing on solid ground. The video then transitions to show two other men, one in a shirt and the other in a blue shirt, also standing on the ground. The clip concludes with a black screen featuring a large black bear and a big bird.

Fig. 16. Question and answer about clips from *Game of Thrones*, which tells the epic fantasy tale of power struggles and political intrigue among the Seven Kingdoms, entwined with intricate family relationships, all set against the backdrop of an ancient, mystical threat.

with the same question, retaining more merged frames for strongly related segments leads to a noticeably elongated long-term memory, where effective positional encoding becomes challenging, reducing comprehension of long videos. This highlights the necessity of balancing the retention of dense information and the compression for effective long-term video understanding.

Judging Relevance Basis. Determining the relevance of short-term memory to a question based on similarity can be approached in three ways: comparing the highest similarity within a segment, the lowest, or the average with the question-frame similarity threshold σ. According to Fig. 12 (bottom right), using the minimum or average similarity shows similar performance. However, selecting the maximum similarity as the criterion leads to a performance drop. We believe this is due to the lenient judgment of relevance between the question and frames when choosing the maximum similarity within a segment, which introduces more redundant information. Thus we have elected to utilize the average similarity as the criterion for comparison.

5.3 Case Study

We perform an extensive case study of MovieChat on a variety of open-ended long video (such as cartoon movie and TV series) including the ■ breakpoint mode (Q#1) and the ■ global mode (Q#2). The evaluation is conducted between our approach and previous methods [41], [48], [93] as shown in Fig. 13. For Q#1 in breakpoint mode, we mark the timestamp when the question is asked. For long videos over 10K frames, MovieChat is still capable of providing excellent responses to questions regarding both the current moment and the entire video content with less hallucination. We also provide more examples to show the long video scene understanding and temporal understanding ability of MovieChat in Fig. 14, 15 and 16.

6 Limitation

Although MovieChat has demonstrated impressive abilities in long video understanding, it is still an early-stage prototype and has some limitations, including 1) Limited perception capacities. The performance of our approach is hindered by the pre-trained short video understanding model. 2) Inadequate Time Processing. MovieChat provides only rough estimates of the duration proportions of events within long videos, lacking precision in temporal details.

7 Conclusion

Conclusively, we present an innovative video understanding system integrating video foundation models and large language models. By incorporating an enhanced memory mechanism represented by tokens in Transformers, our proposed system, MovieChat overcomes challenges associated with analyzing long videos. MovieChat achieves state-of-the-art performance in long video understanding, surpassing existing systems, which are limited to handling videos with few frames.

References

[1] 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. 1, 2

[2] Roy Allen, Peter Mcgeorge, David G Pearson, and Alan Milne. Multiple-target tracking: A role for working memory? *Quarterly journal of experimental psychology*, 59(6):1101–1116, 2006. 3

[3] Norah Alzahrani, Hisham Abdullah Alyahya, Yazeed Alnumay, Sultan Alrashed, Shaykhah Alsubaie, Yusef Almushaykeh, Faisal Mirza, Nouf Alotaibi, Nora Altwairesh, Areeb Alowisheq, et al. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. *arXiv preprint arXiv:2402.01781*, 2024. 9

[4] Anthropic. Meet claude, 2023. 8, 9, 10

[5] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 4

[6] Richard C Atkinson and Richard M Shiffrin. Chapter: Human memory: A proposed system and its control processes. *The psychology of learning and motivation*, 2:89–195, 1968. 2

[7] George Awad, Asad A Butt, Keith Curtis, Jonathan Fiscus, Afzal Godil, Yooyoung Lee, Andrew Delgado, Jesse Zhang, Eliot Godard, Baptiste Chocot, et al. Trecvid 2020: A comprehensive campaign for evaluating video retrieval tasks across multiple application domains. *arXiv preprint arXiv:2104.13473*, 2021. 3

[8] George Awad, Asad A Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzal Godil, Andrew Delgado, Jesse Zhang, Eliot Godard, Lukas Diduch, et al. Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval. *arXiv preprint arXiv:2009.09984*, 2020. 3

[9] George Awad, Asad A Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzad Godil, David Joy, Andrew Delgado, Alan F Smeaton, Yvette Graham, et al. Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search. In *Proceedings of TRECVID 2018*, 2018. 3

[10] George Awad, Asad A Butt, Jonathan Fiscus, David Joy, Andrew Delgado, Willie Mcclinton, Martial Michel, Alan F Smeaton, Yvette Graham, Wessel Kraaij, et al. Trecvid 2017: evaluating ad-hoc and instance video search, events detection, video captioning, and hyperlinking. In *TREC Video Retrieval Evaluation (TRECVID)*, 2017. 3

[11] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. 3

[12] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *CoRR*, abs/2104.00650, 2021. 7

[13] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *The Eleventh International Conference on Learning Representations*, 2022. 5

[14] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1, 2

[15] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: multi-object tracking with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8090–8100, 2022. 3

[16] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 3

[17] Wenhao Chai and Gaoang Wang. Deep vision multimodal learning: Methodology, benchmark, and trend. *Applied Sciences*, 12(13):6588, 2022. 1

[18] David 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: human language technologies*, pages 190–200, 2011. 3

[19] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory

model. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII*, pages 640–658. Springer, 2022. 3

[20] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023. 1, 2

[21] StableLM contributors. Stablevicuna. 2023. 10

[22] 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. *arXiv preprint arXiv:2305.06500*, 2023. 1, 2

[23] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 1

[24] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. 2022. 4

[25] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Zhenyu Qiu, Wei Lin, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 1

[26] Difei Gao, Luowei Zhou, Lei Ji, Linchao Zhu, Yi Yang, and Mike Zheng Shou. Mist: Multi-modal iterative spatial-temporal transformer for long-form video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14773–14783, 2023. 3

[27] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 1, 2

[28] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023. 2

[29] Tao Gong, Chengqi Lyu, Shilong Zhang, Yudong Wang, Miao Zheng, Qian Zhao, Kuikun Liu, Wenwei Zhang, Ping Luo, and Kai Chen. Multimodal-gpt: A vision and language model for dialogue with humans. *arXiv preprint arXiv:2305.04790*, 2023. 1, 2

[30] Zhicheng Hao, Jun Qiu, Haimiao Zhang, Guangbo Ren, and Chang Liu. Umotma: Underwater multiple object tracking with memory aggregation. *Frontiers in Marine Science*, 9:1071618, 2022. 3

[31] Li Hu, Peng Zhang, Bang Zhang, Pan Pan, Yinghui Xu, and Rong Jin. Learning position and target consistency for memory-based video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4144–4154, 2021. 3

[32] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 709–727. Springer, 2020. 3, 6

[33] hwchase17. langchain. https://github.com/hwchase17/langchain, 2023. 9

[34] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2758–2766, 2017. 6

[35] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019. 5

[36] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 8

[37] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023. 1, 2

[38] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 1, 2

[39] 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. 4

[40] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 1

[41] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 1, 2, 7, 8, 9, 10, 14

[42] Boyu Liu, Yanzhao Wang, Yu-Wing Tai, and Chi-Keung Tang. Mavot: Memory-augmented video object tracking. *arXiv preprint arXiv:1711.09414*, 2017. 3

[43] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 1

[44] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3202–3211, 2022. 4

[45] Chenyang Lyu, Minghao Wu, Longyue Wang, Xinting Huang, Bingshuai Liu, Zefeng Du, Shuming Shi, and Zhaopeng Tu. Macaw-llm: Multi-modal language modeling with image, audio, video, and text integration. *arXiv preprint arXiv:2306.09093*, 2023. 1

[46] Chao Ma, Jia-Bin Huang, Xiaokang Yang, and Ming-Hsuan Yang. Adaptive correlation filters with long-term and short-term memory for object tracking. *International Journal of Computer Vision*, 126:771–796, 2018. 3

[47] Fan Ma, Xiaojie Jin, Heng Wang, Yuchen Xian, Jiashi Feng, and Yi Yang. Vista-llama: Reliable video narrator via equal distance to visual tokens. *arXiv preprint arXiv:2312.08870*, 2023. 2

[48] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023. 1, 2, 7, 8, 9, 10, 14

[49] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *arXiv preprint arXiv:2308.09126*, 2023. 3, 6, 9

[50] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019. 3

[51] openai. Gpt3.5, 2021. 2021. 7, 8, 9, 10

[52] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 2

[53] Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*, 2023. 9

[54] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013. 3

[55] Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele. Coherent multi-sentence video description with variable level of detail. In *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*, pages 184–195. Springer, 2014. 3

[56] Anna Rohrbach, Marcus Rohrbach, Siyu Tang, Seong Joon Oh, and Bernt Schiele. Generating descriptions with grounded and co-referenced people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4979–4989, 2017. 2

[57] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. A database for fine grained activity detection of cooking activities. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1194–1201. IEEE, 2012. 3

[58] Marcus Rohrbach, Michaela Regneri, Mykhaylo Andriluka, Sikandar Amin, Manfred Pinkal, and Bernt Schiele. Script data for attribute-based recognition of composite activities. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I 12*, pages 144–157. Springer, 2012. 3

[59] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer*

Vision, 119:346–373, 2016. 3

[60] Fadime Sener, Dipika Singhania, and Angela Yao. Temporal aggregate representations for long-range video understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 154–171. Springer, 2020. 3

[61] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 629–645. Springer, 2020. 3

[62] Hongje Seong, Seoung Wug Oh, Joon-Young Lee, Seongwon Lee, Suhyeon Lee, and Euntai Kim. Hierarchical memory matching network for video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12889–12898, 2021. 3

[63] Mike Zheng Shou, Stan Weixian Lei, Weiyao Wang, Deepti Ghadiyaram, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8075–8084, 2021. 3, 6

[64] Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba, Chen Zhao, Silvio Giancola, and Bernard Ghanem. Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5026–5035, 2022. 3, 6

[65] Larry R Squire, Lisa Genzel, John T Wixted, and Richard G Morris. Memory consolidation. *Cold Spring Harbor perspectives in biology*, 7(8):a021766, 2015. 3

[66] Jianlin Su. Bert position encoding. https://kexue.fm/archives/7947, 2023. 5

[67] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*, 2023. 1

[68] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023. 1, 2

[69] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4631–4640, 2016. 3

[70] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1, 2, 10, 11

[71] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 2, 10, 11

[72] Junke Wang, Dongdong Chen, Chong Luo, Xiyang Dai, Lu Yuan, Zuxuan Wu, and Yu-Gang Jiang. Chatvideo: A tracklet-centric multimodal and versatile video understanding system. *arXiv preprint arXiv:2304.14407*, 2023. 1, 2

[73] Jiahao Wang, Guo Chen, Yifei Huang, Limin Wang, and Tong Lu. Memory-and-anticipation transformer for online action understanding. *arXiv preprint arXiv:2308.07863*, 2023. 3

[74] 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. *arXiv preprint arXiv:2205.14100*, 2022. 8, 9

[75] Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, et al. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*, 2023. 1, 2

[76] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. Longvlm: Efficient long video understanding via large language models. *arXiv preprint arXiv:2404.03384*, 2024. 2

[77] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 284–293, 2019. 2

[78] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021. 2, 6

[79] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022. 3

[80] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786, 2021. 7, 8

[81] Junbin Xiao, Angela Yao, Yicong Li, and Tat Seng Chua. Can i trust your answer? visually grounded video question answering. *arXiv preprint arXiv:2309.01327*, 2023. 6

[82] Ming Xin, Wenjie Sun, Kaifang Li, and Guancheng Hui. Multi-object tracking with spatial-temporal correlation memory networks. In *2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA)*, pages 616–619. IEEE, 2022. 3

[83] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653, 2017. 6, 7

[84] Haiyang Xu et al. mplug-2: A modularized multi-modal foundation model across text, image and video. *arXiv preprint arXiv:2302.00402*, 2023. 8, 9

[85] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. June 2016. 7

[86] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. *Advances in Neural Information Processing Systems*, 35:124–141, 2022. 7, 10

[87] Tianyu Yang and Antoni B Chan. Learning dynamic memory networks for object tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pages 152–167, 2018. 3

[88] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023. 1, 2

[89] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9127–9134, 2019. 7, 8, 10

[90] Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23056–23065, 2023. 3

[91] Kuo-Hao Zeng, Tseng-Hung Chen, Juan Carlos Niebles, and Min Sun. Title generation for user generated videos. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 609–625. Springer, 2016. 3

[92] 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. 1, 2, 7, 8, 9, 10

[93] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 7, 8, 10, 14

[94] Yue Zhao and Philipp Krähenbühl. Real-time online video detection with temporal smoothing transformers. In *European Conference on Computer Vision*, pages 485–502. Springer, 2022. 10

[95] Zhonghan Zhao, Wenhao Chai, Xuan Wang, Li Boyi, Shengyu Hao, Shidong Cao, Tian Ye, Jenq-Neng Hwang, and Gaoang Wang. See and think: Embodied agent in virtual environment. *arXiv preprint arXiv:2311.15209*, 2023. 1

[96] Zhonghan Zhao, Kewei Chen, Dongxu Guo, Wenhao Chai, Tian Ye, Yanting Zhang, and Gaoang Wang. Hierarchical auto-organizing system for open-ended multi-agent navigation. *arXiv preprint arXiv:2403.08282*, 2024. 1

[97] Zhonghan Zhao, Ke Ma, Wenhao Chai, Xuan Wang, Kewei Chen, Dongxu Guo, Yanting Zhang, Hongwei Wang, and Gaoang Wang. Do we really need a complex agent system? distill embodied agent into a single model. *arXiv preprint arXiv:2404.04619*, 2024. 1

[98] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. On large language models' selection bias in multi-choice questions. *arXiv preprint arXiv:2309.03882*, 2023. 9

[99] Xingyi Zhou, Anurag Arnab, Shyamal Buch, Shen Yan, Austin Myers, Xuehan Xiong, Arsha Nagrani, and Cordelia Schmid. Streaming dense video captioning. *arXiv preprint arXiv:2404.01297*, 2024. 3, 10

[100] Zechu Zhou, Xinyu Zhou, Zhaoyu Chen, Pinxue Guo, Qian-Yu Liu, and Wenqiang Zhang. Memory network with pixel-level spatio-temporal learning for visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 3

[101] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2