VideoAgent: Long-form Video Understanding with Large Language Model as Agent

Xiaohan Wang*, Yuhui Zhang*, Orr Zohar, and Serena Yeung-Levy

Stanford University
{xhanwang,yuhuiz,orrzohar,syyeung}@stanford.edu

Abstract. Long-form video understanding represents a significant challenge within computer vision, demanding a model capable of reasoning over long multi-modal sequences. Motivated by the human cognitive process for long-form video understanding, we emphasize interactive reasoning and planning over the ability to process lengthy visual inputs. We introduce a novel agent-based system, *VideoAgent*, that employs a large language model as a central agent to iteratively identify and compile crucial information to answer a question, with vision-language foundation models serving as tools to translate and retrieve visual information. Evaluated on the challenging EgoSchema and NExT-QA benchmarks, *VideoAgent* achieves 54.1% and 71.3% zero-shot accuracy with only 8.4 and 8.2 frames used on average. These results demonstrate superior effectiveness and efficiency of our method over the current state-of-the-art methods, highlighting the potential of agent-based approaches in advancing long-form video understanding.

Keywords: Long-form Video Understanding · Large Language Model Agent · Vision-Language Foundation Models

1 Introduction

Understanding long-form videos, which range from minutes to hours, poses a significant challenge in the field of computer vision. This task demands a model capable of processing multi-modal information, handling exceedingly long sequences, and reasoning over these sequences effectively.

Despite numerous attempts [10,12,15,28,31,43,45,52,55,61] to address this challenge by enhancing these capabilities, existing models struggle to excel in all three areas simultaneously. Current large language models (LLMs) excel in reasoning and handling long contexts [14,49,54,64], yet they lack the capability to process visual information. Conversely, visual language models (VLMs) struggle to model lengthy visual inputs [9,18,19,22,24]. Early efforts have been made to enable VLMs' long context modeling capability, but these adaptations underperform in video understanding benchmarks and are inefficient in dealing with long-form video content [23].

* Equal contribution. Project page: https://wxh1996.github.io/VideoAgent-Website/

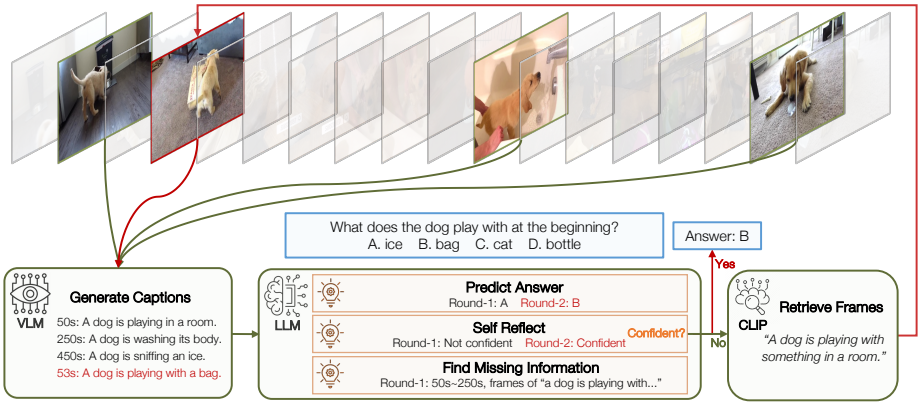

Fig. 1: *Overview of VideoAgent.* Given a long-form video, *VideoAgent* iteratively searches and aggregates key information to answer the question. The process is controlled by a large language model (LLM) as the agent, with the visual language model (VLM) and contrastive language-image model (CLIP) serving as tools.

Do we really need to feed the entire long-form video directly into the model? This diverges significantly from how humans achieve the long-form video understanding task. When tasked with understanding a long video, humans typically rely on the following interactive process to formulate an answer: The process begins with a quick overview of the video to understand its context. Subsequently, guided by the specific question at hand, humans iteratively select new frames to gather relevant information. Upon acquiring sufficient information to answer the question, the iterative process is concluded, and the answer is provided. Throughout this process, the reasoning capability to control this iterative process is more critical than the capacity to directly process lengthy visual inputs.

Drawing inspiration from how humans understand long-form videos [13,50], we present VideoAgent, a system that simulates this process through an agent-based system. We formulate the video understanding process as a sequence of states, actions, and observations, with an LLM serving as the agent controlling this process (Figure 1). Initially, the LLM familiarizes itself with the video context by glancing at a set of uniformly sampled frames from the video. During each iteration, the LLM assesses whether the current information (state) is sufficient to answer the question; if not, it identifies what additional information is required (action). Subsequently, it utilizes CLIP [37] to retrieve new frames containing this information (observation) and VLM to caption these new frames into textual descriptions, updating the current state. This design emphasizes the reasoning capability and iterative processes over the direct processing of long visual inputs, where the VLM and CLIP serve as instrumental tools to enable the LLM to have visual understanding and long-context retrieval capabilities.

Our work differs from previous works in two aspects. Compared to the works that uniformly sample frames or select frames in a single iteration [17,58,68], our method selects frames in a multi-round fashion, which ensures the information

gathered to be more accurate based on the current need. Compared to the works that retrieve frames using the original question as the query [58,68], we rewrite the query to enable more accurate and fine-grained frame retrieval.

Our rigorous evaluation of two well-established long-form video understanding benchmarks, EgoSchema [29] and NExT-QA [57], demonstrates *VideoAgent*'s exceptional effectiveness and efficiency compared to existing methods. *VideoAgent* achieves 54.1% and 71.3% accuracy on these two benchmarks, respectively, outperforming concurrent state-of-the-art method LLoVi [69] by 3.8% and 3.6%. Notably, *VideoAgent* only utilizes 8.4 frames on average to achieve such performance, which is 20x fewer compared to LLoVi. Our ablation studies highlight the significance of the iterative frame selection process, which adaptively searches and aggregates relevant information based on the complexity of the videos. Additionally, our case studies demonstrate that *VideoAgent* generalizes to arbitrarily long videos, including those extending to an hour or more.

In summary, *VideoAgent* represents a significant stride for long-form video understanding, which embraces the agentic system to mimic human cognitive process and underscores the primacy of reasoning over long-context visual information modeling. We hope our work not only sets a new benchmark in long-form video understanding but also sheds light on future research in this direction.

2 Related Work

2.1 Long-form Video Understanding

Long-form video understanding is a particularly challenging domain in computer vision due to the inherent complexity and high dimensionality of spatio-temporal inputs, which leads to significant computational demands. Long-form video understanding methods need to balance computational efficiency and performance, and can broadly be categorized into selective or compressive sparsity strategies.

Compressive sparsity methods [10,12,15,31,43,45,52,55,61,67], attempt to compress the video into meaningful embeddings/representations with the minimum possible dimensionality. For example, MovieChat [43] employs a memory consolidation mechanism that merges similar adjacent frame tokens based on cosine similarity, effectively reducing token redundancy in long video sequences. Chat-UniVi [15] utilized kNN clustering to spatio-temporally compress video tokens. However, the compression need not happen on the embeddings themselves, and can be compressed into space-time graphs [10,52,61] or even text [20,39,69]. For example, Zhang et. al. [69] introduced LLoVi, and have shown that simply captioning videos before and prompting an LLM with these captions can serve as a strong baseline.

Meanwhile, selective-compressive methodologies attempt to sub-sample the video into more meaningful frames, utilizing the question/text as a guide, and in effect attempt to only sample frames relevant to the question at hand [7,21,38,58,68]. For instance, methods such as R-VLM and R2A [8,34,58] utilize a CLIP model to retrieve relevant frames given a text prompt, and while Q-ViD [39]

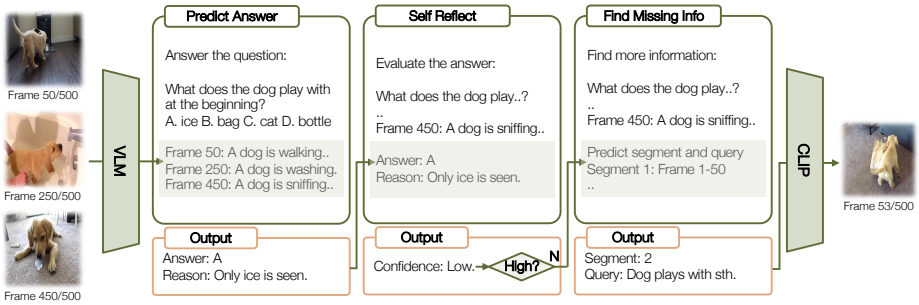

Fig. 2: *Detailed view of VideoAgent's iterative process.* Each round starts with the state, which includes previously viewed video frames. The large language model then determines subsequent actions by answering prediction and self-reflection. If additional information is needed, new observations are acquired in the form of video frames.

utilizes the question to selectively caption the video. Unlike previous works, we allow the LLM to direct the video frames to be sampled by the captioner.

2.2 LLM Agents

An agent is defined as an entity that makes decisions and takes actions in a dynamic, real-time environment to achieve some specific goals. Advances in large language models (LLMs), especially their emerging reasoning and planning capabilities [54, 64, 72], has inspired recent research in natural language processing to leverage them as agents in real-world scenarios [36, 65]. These models have demonstrated great success across various scenarios, such as online search, card game playing, and database management [26, 27, 63]. Their effectiveness is further amplified with methods such as chain-of-thought reasoning or self-reflection [42, 54].

Simultaneously, the computer vision community has begun to explore LLM-as-agent-based approach in diverse visual contexts, such as GUI understanding and robot navigation [3, 5, 9, 46]. In the area of long-form video understanding, several studies have made an initial attempt with an agent-like approach, which utilize LLMs to interact with external tools or to incorporate additional functionalities [6, 46, 62]. Unlike these approaches, our work reformulates video understanding as a decision-making process, which is inspired by how humans solve video interpretation methods. We view the video as an environment where decisions involve either seeking more information or concluding the interaction. This perspective has guided the creation of *VideoAgent*, a novel framework that significantly diverges from existing methodologies by emphasizing the decision-making aspects inherent in video understanding.

3 Method

In this section, we introduce the method of *VideoAgent*. *VideoAgent* is inspired by the human cognitive process of understanding long-form videos. Given a video

with the question, a human will first glance at several frames to understand its context, then iteratively search additional frames to obtain enough information to answer the question, and finally aggregate all the information and make the prediction.

We formulate the process into a sequence of states, actions, and observations $\{(s_t, a_t, o_t)|1 \leq t \leq T\}$, where the state is the existing information of all the seen frames, action is whether to answer the question or continue to search new frames, observation is the new frames seen in the current iteration, and T is the maximum number of iterations.

We leverage large language model (LLM) GPT-4 [32] as an agent to perform the above process (Figure 1). LLMs have been demonstrated to have memory, reasoning and planning, and tool-use capability [40,47,54,72], which can be used to model states, actions, and observations, respectively.

3.1 Obtaining the Initial State

To start the iterative process, we first familiarize the LLM with the context of the video, which can be achieved by glancing at N frames uniformly sampled from the video. Since the LLM has no capability for visual understanding, we leverage vision-language models (VLMs) to convert the visual content to language descriptions. Specifically, we caption these N frames with the prompt "describe the image in detail" and feed the captions to the LLM. This initial state s_1 records a sketch of the content and semantics of the video.

3.2 Determining the Next Action

Given the current state s_t that stores the information of all the seen frames, there are two possible options for the next action a_t:

- **Action 1: answer the question.** If the information in state s_t is enough to answer the question, we should answer the questions and exit the iterative process.
- **Action 2: search new information.** If the current information in s_t is insufficient, we should decide what further information is required to answer the question and continue searching for it.

To decide between actions 1 and 2, we need the LLM to reason over the question and existing information. This is achieved by a three-step process. First, we force the LLM to make a prediction based on the current state and question via chain-of-thought prompting. Second, we ask the LLM to self-reflect and generate a confidence score based on the state, question, prediction and its reasoning process generated by step 1. The confidence score has three levels: 1 (insufficient information), 2 (partial information), and 3 (sufficient information). Finally, we choose action 1 or 2 based on the confidence score. This process is illustrated in Figure 2. We propose to use a three-step process over a single-step process that directly chooses action as direct prediction always decides to search for new information (Action 2). This self-reflection process is motivated by [42], which has demonstrated superior effectiveness in natural language processing.

3.3 Gathering a New Observation

Suppose the LLM determines insufficient information to answer the question and chooses to search for new information. In that case, we further ask the LLM to decide what extra information is needed so that we can leverage tools to retrieve them (Figure 2). Since some piece of information could occur multiple times within a video, we perform segment-level retrieval instead of video-level retrieval to enhance the temporal reasoning capability. For example, suppose the question is "What is the toy left on the sofa after the boy leaves the room?" and that we have seen the boy leave the room at frame i. If we retrieve with the query "a frame showing the toy on the sofa," there may be frames before frame i containing "toy on the sofa", but they are irrelevant to answering the question.

To perform segment-level retrieval, we first split the video into different segments based on the seen frame indices, and ask the LLM to predict what segments to retrieve with the query texts. For example, if we have seen frames i, j, and k of a video, one valid prediction is segment 2 (frame i to j) with the query "a frame showing the toy on the sofa".

We leverage CLIP [37] to obtain this additional information given the output by the LLM. Specifically, given each query and segment, we return the image frame with the highest cosine similarity with the text query in that segment. These retrieved frames are served as observations to update the state.

The use of CLIP in the retrieval step is computationally efficient and negligible compared to using an LLM or VLM for several reasons. Firstly, CLIP's feature computation involves just a single feed-forward process. Secondly, CLIP employs an image-text late interaction architecture, enabling the caching and reusing of image frame features across different text queries. Lastly, our segment-level retrieval design only requires computing features within specific segments, further enhancing efficiency. Empirically, our experiments show that CLIP computations are less than 1% of that of a VLM and LLM.

3.4 Updating the Current State

Finally, given the new observations (i.e., retrieved frames), we leverage VLMs to generate captions for each frame, and then simply sort and concatenate the new captions with old frame captions based on frame index, and ask the LLM to generate next-round predictions.

A question one may posit is why we leverage the multi-round process, as some existing works use all or uniformly sampled frames as the state in a single step [17, 69]. There are many advantages of our approach over these baselines. First, using too many frames introduces extensive information and noise, which leads to performance degradation because LLMs suffer from long contexts and can be easily distracted [25, 41]. Furthermore, it is computationally inefficient and hard to scale up to hour-long videos due to the LLM context length limit [32]. On the opposite, using too few frames may not capture relevant information. Our adaptive selection strategy finds the most relevant information and requires the lowest cost to answer questions at different difficulty levels.

We summarize the *VideoAgent* as Algorithm 1.

Algorithm 1 *VideoAgent*

Require: Video v, question q, LLM F_l, VLM F_v, CLIP F_c, max iteration T, confidence threshold C
Ensure: Prediction $\hat{y}$, state-action-observation sequence $\{s_t, a_t, o_t | 1 \leq t \leq T\}$

$s_1 \leftarrow \texttt{GenerateCaptions}(F_v, \texttt{UniformSample}(v))$
for $t = 1$ **to** T **do**
  $\hat{y} \leftarrow \texttt{PredictAnswer}(F_l, s_t, q)$
  $c \leftarrow \texttt{SelfReflect}(F_l, s_t, q, \hat{y})$
  if $a_t \leftarrow \mathbb{1}_{[c \geq C]}$ **then**
    break
  else
    $h \leftarrow \texttt{FindMissingInfo}(F_l, s_t, q)$
    $o_t \leftarrow \texttt{RetrieveFrames}(F_c, v, h)$
    $s_{t+1} \leftarrow \texttt{Merge}(s_t, \texttt{GenerateCaptions}(F_v, o_t))$
  end if
end for
return $\hat{y}$, $\{s_t, a_t, o_t | 1 \leq t \leq T\}$

4 Experiments

In this section, we first introduce the datasets and implementation details, and then we present the results, analyses, ablations, and case studies of *VideoAgent*.

4.1 Datasets and Metrics

In our experiments, we use two distinct well-established datasets to benchmark our model's performance, with a particular focus on zero-shot understanding capabilities.

EgoSchema [29]. EgoSchema is a benchmark for long-form video understanding, featuring 5,000 multiple-choice questions derived from 5,000 egocentric videos. These videos provide an egocentric viewpoint of humans engaged in a wide range of activities. A distinctive feature of this dataset is the length of its videos, each lasting 3 minutes. EgoSchema comprises only a test set, with a subset of 500 questions having publicly available labels. The full set of questions is evaluated exclusively on the official leaderboard.

NExT-QA [57]. The NExT-QA dataset includes 5,440 natural videos that feature object interactions in daily life, accompanied by 48,000 multiple-choice questions. The average length of video is 44 seconds. These questions fall into three categories: Temporal, Causal, and Descriptive, providing a comprehensive evaluation for video understanding models. In line with standard practices, our zero-shot evaluation focused on the validation set, which contains 570 videos and 5,000 multiple-choice questions. We additionally follow [4] to report performance on the ATP-hard subset of the NExT-QA validation set. This subset

Method		Frames	Subset	**Full**
FrozenBiLM [60]	[NeurIPS2022]	90	-	26.9
InternVideo [53]	[arXiv2022.12]	90	-	32.1
ImageViT [35]	[arXiv2023.12]	16	40.8	30.9
ShortViViT$_{loc}$ [35]	[arXiv2023.12]	32	49.6	31.3
LongViViT [35]	[arXiv2023.12]	256	56.8	33.3
SeViLA [68]	[NeurIPS2023]	32	25.7	22.7
Vamos [51]	[arXiv2023.11]	-	-	48.3
LLoVi [69]	[arXiv2024.2]	180	57.6	50.3
MC-ViT-L [2]	[arXiv2024.2]	128+	62.6	44.4
VideoAgent (`ours`)		**8.4**	60.2	**54.1**

Table 1: *Results on EgoSchema compared to public models.* Full-set results are obtained from the official leaderboard.

Model		Subset	**Full**
Random Chance		20.0	20.0
Bard only (blind) [2]	[2023.3]	27.0	33.2
Bard + ImageViT [35]	[2023.3]	35.0	35.0
Bard + ShortViViT [35]	[2023.3]	42.0	36.2
Bard + PALI [35]	[2023.3]	44.8	39.2
GPT-4 Turbo (blind) [2]	[2023.4]	31.0	30.8
GPT-4V [2]	[2023.9]	63.5	55.6
Gemini 1.0 Pro [48]	[2023.12]	-	55.7
VideoAgent (`ours`)		60.2	54.1

Table 2: *Results on EgoSchema compared to large-scale proprietary models.*

keeps the hardest QA pairs that can not be solved with one frame, focusing more on long-term temporal reasoning.

Since each dataset features multiple-choice questions, we utilized accuracy as our evaluation metric.

4.2 Implementation Details

We decode all the videos in our experiments at 1 fps and use EVA-CLIP-8B-plus [44] to retrieve the most relevant frames based on the cosine similarity between the generated descriptions and the frame features. For the experiments on EgoSchema, we utilize LaViLa [70] as the captioner, which is a clip-based captioning model. Following [69], to ensure zero-shot evaluation, we utilize the LaViLa model retrained on the ego4D data, filtering out the overlapped videos with EgoSchema. We sample the video clip for captioning based on the frame index returned by the CLIP retrieval module. For NExT-QA, we utilize CogAgent [9] as the captioner. We use GPT-4 [32] as the LLM for all experiments, the version of GPT is fixed to *gpt-4-1106-preview* to ensure reproducibility.

4.3 Comparison with State-of-the-arts

VideoAgent sets new benchmarks, achieving state-of-the-art (SOTA) results on the EgoSchema and NExT-QA datasets, surpassing previous methods significantly while requiring only a minimal number of frames for analysis.

EgoSchema. As shown in Tables 1 and 2, *VideoAgent* achieves an accuracy of 54.1% on the EgoSchema full dataset and 60.2% on a 500-question subset. The full dataset's accuracy was verified by uploading our model's predictions to the official leaderboard, as ground-truth labels are not publicly available. These results not only significantly outperform the previous SOTA method LLoVi [69] by 3.8%, but also achieve comparable performance to advanced proprietary models like Gemini-1.0 [48]. Notably, our method requires an average of only 8.4 frames per video — significantly fewer by 2x to 30x compared to existing approaches.

Methods		Val				ATP-hard subset		
		Acc@C	Acc@T	Acc@D	Acc@All	Acc@C	Acc@T	Acc@All
Supervised								
VFC [59]	[ICCV2021]	49.6	51.5	63.2	52.3	-	-	-
ATP [4]	[CVPR2022]	53.1	50.2	66.8	54.3	38.4	36.5	38.8
MIST [7]	[CVPR2023]	54.6	56.6	66.9	57.2	-	-	-
GF [1]	[NeurIPS2023]	56.9	57.1	70.5	58.8	48.7	50.3	49.3
CoVGT [56]	[TPAMI2023]	59.7	58.0	69.9	60.7	-	-	-
SeViT [16]	[arXiv2023.1]	54.0	54.1	71.3	56.7	43.3	46.5	-
HiTeA [66]	[ICCV2023]	62.4	58.3	75.6	63.1	47.8	48.6	-
Zero-shot								
VFC [30]	[ICCV2023]	51.6	45.4	64.1	51.5	32.2	30.0	31.4
InternVideo [53]	[arXiv2022.12]	43.4	48.0	65.1	49.1	-	-	-
AssistGPT [6]	[arXiv2023.6]	60.0	51.4	67.3	58.4	-	-	-
ViperGPT [46]	[ICCV2023]	-	-	-	60.0	-	-	-
SeViLA [68]	[NeurIPS2023]	61.3	61.5	75.6	63.6	-	-	-
LLoVi [69]	[arXiv2024.2]	69.5	61.0	75.6	67.7	-	-	-
VideoAgent (`ours`)		**72.7**	**64.5**	**81.1**	**71.3**	**57.8**	**58.8**	**58.4**

Table 3: *Results on NExT-QA compared to the state of the art.* C, T, and D are causal, temporal, and descriptive subsets, respectively.

NExT-QA. In Table 3, we show that *VideoAgent* achieves a 71.3% accuracy on the NExT-QA full validation set, surpassing the former SOTA, LLoVi [69], by 3.6%. With an average of merely 8.2 frames used per video for zero-shot evaluation, *VideoAgent* consistently outperforms previous supervised and zero-shot methods across all subsets by a large margin, including those testing the model's causal, temporal, and descriptive abilities. Importantly, *VideoAgent* achieves remarkable performance improvements on the more challenging subsets, ATP-hard [4], demonstrating its adeptness at addressing complex long-form video queries.

These results underscore *VideoAgent*'s exceptional effectiveness and efficiency in processing and understanding complex questions from long-form videos.

4.4 Analysis of Iterative Frame Selection

One of the key components of *VideoAgent* is its iterative frame selection, which dynamically searches for and aggregates more information until it is sufficient to answer the question, mimicking the human process of understanding videos. We conducted comprehensive analyses and ablation studies to understand this process better.

Frame efficiency. Our first analysis focused on whether frame selection effectively identifies the informative frames needed to answer a question. This can be measured by frame efficiency: given a fixed number of frames, what model accuracy can be achieved. The hypothesis is that the more informative frames it identifies, the higher the frame efficiency should be. In Figure 3 (left), we plot the accuracy of our method compared to uniform sampling baselines and other previous methods on the EgoSchema 500-question subset. Our method significantly outperforms uniform selection and other baselines at the same number of frames, demonstrating its superiority in frame efficiency. Notably, our method,

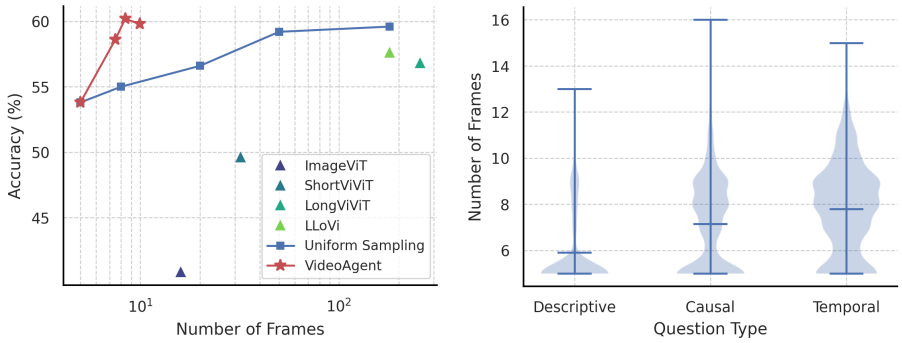

Fig. 3: *(Left) Frame efficiency compared to uniform sampling and previous methods.* X-axis is in log scale. Our method achieves exceptional frame efficiency for long-form video understanding. *(Right) Number of frames for different types of NExT-QA questions.* Min, mean, max, distribution are plotted. *VideoAgent* selects more frames on questions related to temporal reasoning than causal reasoning and descriptive questions.

which uses only 8.4 frames to achieve 60.2% accuracy, surpasses the baseline that uniformly samples 180 frames to achieve 59.6% accuracy. This underscores the effectiveness of our method in finding informative frames and reveals that more frames do not always lead to better performance, as irrelevant and noisy information can overwhelm the language model with long contexts and distractions [25, 41].

Number of rounds. We also analyzed how the number of iterative rounds affects model performance. In the same Figure 3 (left), we plot the performance across 1-4 rounds and the number of selected frames, achieving accuracies of 53.8%, 58.6%, 60.2%, and 59.8% with 5, 7.5, 8.4, and 9.9 frames, respectively. The performance improves with additional rounds but saturates at three rounds on the EgoSchema 500-question subset. This indicates that our approach can efficiently find the information needed to answer the question, and beyond a certain point, additional information does not further help in answering the question.

Different question types. Given that our frame selection process is dynamic, with the language model agent determining whether the information is sufficient, we hypothesized that different question types might require varying amounts of information due to their differing levels of difficulty. We tested this hypothesis on the NExT-QA dataset, which provides annotations for each question type: descriptive tasks, causal reasoning, or temporal reasoning. In Figure 3 (right), we plot the distribution of the number of frames for each type of question. We observed that the average number of frames used ranks as follows: descriptive (5.9 frames), causal (7.1 frames), and temporal (7.8 frames) questions. This aligns with human intuition that descriptive tasks often require fewer frames as initial uniform sampling is usually sufficient, whereas reasoning tasks, especially temporal reasoning, require viewing more frames to accurately answer the question.

Uniform	Uni-7	Uni-9	Uni-11
	54.6	54.8	55.8
Ours	3→6.4	5→8.4	8→11.0
	58.4	**60.2**	**57.4**

Table 4: *Ablation of initial number of uniformly sampled frames.*

Method	Frames	Acc
Ours w/o Seg. Selection	7.5	56.6
Ours w/o Self-Evaluation	11.8	59.6
Ours	8.4	**60.2**

Table 5: *Ablation of segment selection and self-evaluation.*

Initial Number of Frames. Before initiating the iterative frame selection process, we uniformly sample N frames to acquaint the language model with the video context. To explore how the number of initially sampled frames influences model performance and the average number of frames utilized, we conduct an ablation study. Specifically, we sample 3, 5, and 8 frames initially on the EgoSchema 500-question subset and report the findings in Table 4. The results indicate accuracies of 58.4%, 60.2%, and 57.4% with an average of 6.4, 8.4, and 11.0 frames used, respectively. Starting with 5 frames leads to the highest performance. Furthermore, when comparing our method against uniform sampling with a similar or slightly higher number of frames, we observe accuracies of 54.6%, 54.8%, and 55.8% for 7, 9, and 11 frames, respectively. This comparison again highlights the superior efficiency of our frame selection method.

Self-evaluation. During the iterative selection process, we perform a self-evaluation to ascertain whether the available information suffices to answer the query. If sufficient, the iteration terminates at this stage. We benchmark this against a baseline method without self-evaluation, where every question is processed through three rounds of iteration. As detailed in Table 5, we observe an increase in the average number of frames from 8.4 to 11.8 and a decrease in accuracy from 60.2% to 59.6%. These results underscore the efficacy of self-evaluation in determining the adequacy of information, thereby curtailing unnecessary iterations. Notably, gathering more information through additional rounds does not lead to performance improvement but rather results in a marginal decline.

Segment selection. When it is determined that additional information is required, the input videos are divided into segments. The language model then generates queries specifically tailored to retrieve information within those segments. This approach is contrasted with an alternative strategy that involves generating queries without specifying segments. In Table 5, we observe a 3.6% accuracy degradation when segment selection is disabled. Segment selection improves the model's temporal reasoning capabilities and mitigates the risk of conflating information from disparate segments. This is particularly beneficial for queries such as "what happens after...?", where retrieval is only desired from subsequent segments.

4.5 Ablation of Foundation Models

Given that *VideoAgent* integrates three foundational model types — large language model (LLM), visual language model (VLM), and contrastive language-

LLM	Model Size	Acc. (%)
Mistral-8x7B	70B	37.8
Llama2-70B	70B	45.4
GPT-3.5	N/A	48.8
GPT-4	N/A	**60.2**

Table 6: *LLM ablation.*

Captioner	Type	# Words	Acc. (%)
BLIP-2	Frame-based	8.5	52.4
LaViLa	Clip-based	7.2	60.2
CogAgent	Frame-based	74.2	**60.8**

Table 7: *VLM ablation.*

CLIP	Model Size	Resolution	Acc. (%)
OpenCLIP ViT-G	1B	224	59.2
EVA-CLIP-8B	8B	224	59.4
EVA-CLIP-8B-plus	8B	448	**60.2**

Table 8: *CLIP ablation.*

image model (CLIP) — we conduct a series of ablation studies to evaluate the impact of each component's design on the overall performance of the model.

LLM. We initiated our study by evaluating how different LLMs influence the performance of our model, given the pivotal role of LLMs in our methodology, where they function as agents orchestrating the entire process. In Table 6, we compare several state-of-the-art public and proprietary LLMs, including LLaMA-2-70B [49], Mixtral-8x7B [14], GPT-3.5 [33], and GPT-4 [32]. Our findings indicate that GPT-4 significantly outperforms its counterparts. However, it is primarily due to its capability in structured prediction. The iterative process employs JSON for output, where accurate JSON parsing is crucial. GPT-4 demonstrates robust performance in generating correct JSON formats, a feat not as consistently achieved by other models, which remains an active research area in LLM [71].

VLM. Leveraging GPT-4, a text-only model without visual capabilities, we translate image frames into descriptive captions through VLMs, subsequently feeding these captions to GPT-4. To assess the impact of caption quality produced by various VLMs, we examined three state-of-the-art VLMs: frame-based BLIP-2 [19] and CogAgent [9], along with clip-based LaViLa [70] as presented in Table 7. Our analysis revealed that captions from CogAgent and LaViLa yield similar performances, even though their generated captions have significantly different lengths, while BLIP-2 generated captions are much worse.

CLIP. CLIP excels in retrieval tasks due to the late-interaction design for image and text features, eliminating the need for recomputing image embeddings for varying queries. We evaluated three versions of CLIP: OpenCLIP ViT-G [11], EVA-CLIP-8B [44], and EVA-CLIP-8B-plus [44], with the results shown in Table 8. Our findings suggest comparable performance across different models, indicating that retrieval does not constitute a bottleneck for our methodology.

It's important to note that the main contribution of our research is the introduction of a framework emulating the human process of understanding long-form videos, rather than the employment of any specific model. With the rapid developments of foundational models such as LLMs, VLMs, and CLIPs, our approach can be further improved with the integration of better models, or by adopting a

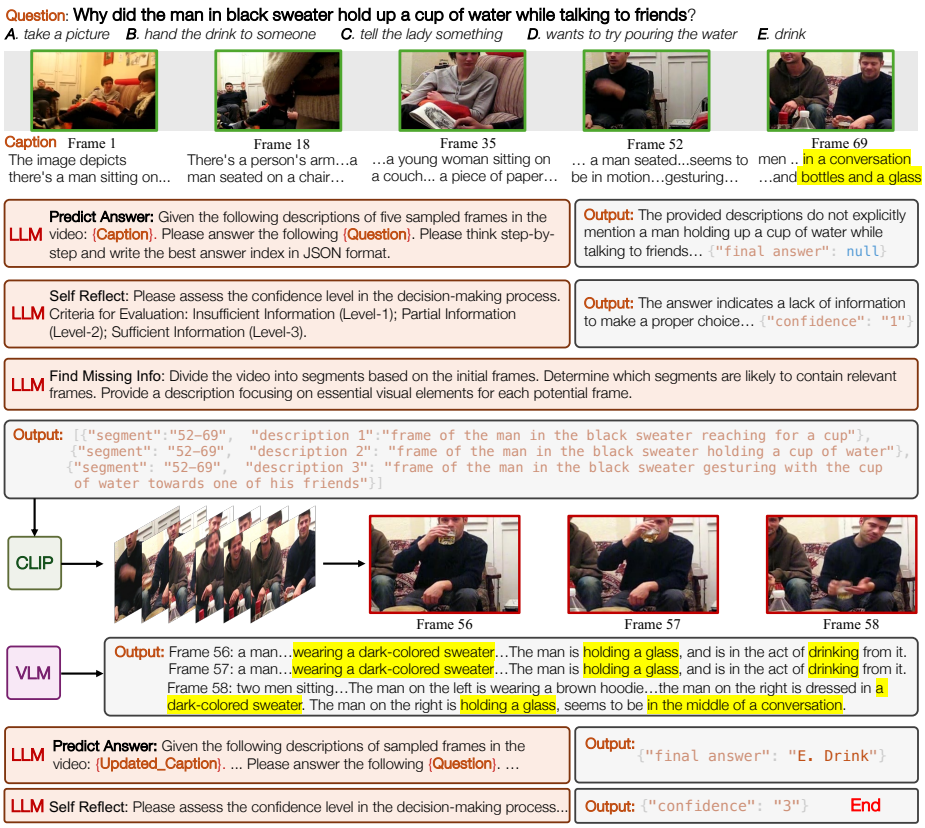

Fig. 4: *Case study on NExT-QA. VideoAgent* accurately identifies missing information in the first round, bridges the information gap in the second round, and thereby makes the correct prediction.

caption-free methodology by replacing GPT-4 with GPT-4V. We hope our work sheds light on future work in this direction.

4.6 Case Studies

We present several case studies to demonstrate the capability of *VideoAgent* in understanding long-form videos.

Questions from NExT-QA [57]. In Figure 4, we illustrate an instance from NExT-QA solved in two iterations. The question is asking why the man holds up a cup of water when talking to friends. *VideoAgent* accurately identify missing information (although the cup is visible in frame 69, it does not reveal the man is holding it). It then determines what additional information is required (frame of the man in the black sweater holding a cup of water). Finally, it utilizes CLIP to retrieve this detail (the man is holding a glass and is in the act of drinking from it) and feel confident about its answer.

Fig. 5: *Case study on hour-long videos. VideoAgent* accurately identifies the key frame during the second iteration, subsequently making an accurate prediction. Conversely, GPT-4V, when relying on 48 uniformly sampled frames up to its maximum context length, does not get successful prediction. However, by integrating the frame pinpointed by *VideoAgent*, GPT-4V is able to correctly answer the question.

Hour-long videos. Given that both NExT-QA and EgoSchema videos span only a few minutes, Figure 5 shows how *VideoAgent* can accurately solve hour-long videos from YouTube[†]. The question is about figuring out the color of the stairs surrounding by green plants, which only occupy a small portion of the video. *VideoAgent* efficiently identifies the necessary information and answers questions within only two iterations and seven frames, outperforming state-of-the-art models like GPT-4V. Notably, GPT-4V struggles with uniform sampling across its maximum context length of 48 images. However, when GPT-4V is provided with the frame pinpointed by *VideoAgent*, it can successfully answer the question. This underscores the potential of enhancing GPT-4V's capabilities in video understanding by integrating our approach.

In conclusion, *VideoAgent* is ready to tackle real-world video understanding, surpassing traditional methods reliant on one-round sparse or dense sampling.

5 Conclusion

In this work, we introduce *VideoAgent*, a system that employs a large language model as an agent to mirror the human cognitive process for understanding long-form videos. *VideoAgent* effectively searches and aggregates information through a multi-round iterative process. It demonstrates exceptional effectiveness and efficiency in long-form video understanding, as evidenced by both quantitative and qualitative studies on various datasets.

Acknowledgements. This work is supported by National Science Foundation under Grant No. 2026498, a Stanford HAI-Google Grant, and a Stanford HAI-AIMI grant. Orr Zohar is funded by the Knight-Hennessy Scholar.

[†] https://www.youtube.com/watch?v=H9Y5_X1sEEA

References

1. Bai, Z., Wang, R., Chen, X.: Glance and focus: Memory prompting for multi-event video question answering. Advances in Neural Information Processing Systems **36** (2024)
2. Balažević, I., Shi, Y., Papalampidi, P., Chaabouni, R., Koppula, S., Hénaff, O.J.: Memory consolidation enables long-context video understanding. arXiv preprint arXiv:2402.05861 (2024)
3. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Chen, X., Choromanski, K., Ding, T., Driess, D., Dubey, A., Finn, C., et al.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. arXiv preprint arXiv:2307.15818 (2023)
4. Buch, S., Eyzaguirre, C., Gaidon, A., Wu, J., Fei-Fei, L., Niebles, J.C.: Revisiting the" video" in video-language understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2917–2927 (2022)
5. Driess, D., Xia, F., Sajjadi, M.S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al.: Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 (2023)
6. Gao, D., Ji, L., Zhou, L., Lin, K.Q., Chen, J., Fan, Z., Shou, M.Z.: Assistgpt: A general multi-modal assistant that can plan, execute, inspect, and learn. arXiv preprint arXiv:2306.08640 (2023)
7. Gao, D., Zhou, L., Ji, L., Zhu, L., Yang, Y., Shou, M.Z.: 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. pp. 14773–14783 (2023)
8. Han, W., Chen, H., Kan, M.Y., Poria, S.: Sas video-qa: Self-adaptive sampling for efficient video question-answering (2023)
9. Hong, W., Wang, W., Lv, Q., Xu, J., Yu, W., Ji, J., Wang, Y., Wang, Z., Dong, Y., Ding, M., et al.: Cogagent: A visual language model for gui agents. arXiv preprint arXiv:2312.08914 (2023)
10. Hussein, N., Gavves, E., Smeulders, A.W.: Videograph: Recognizing minutes-long human activities in videos. arXiv preprint arXiv:1905.05143 (2019)
11. Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., Schmidt, L.: Openclip (Jul 2021). https://doi.org/10.5281/zenodo.5143773, https://doi.org/10.5281/zenodo.5143773, if you use this software, please cite it as below.
12. Islam, M.M., Bertasius, G.: Long movie clip classification with state-space video models. In: European Conference on Computer Vision. pp. 87–104. Springer (2022)
13. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(11), 1254–1259 (1998). https://doi.org/10.1109/34.730558
14. Jiang, A.Q., Sablayrolles, A., Roux, A., Mensch, A., Savary, B., Bamford, C., Chaplot, D.S., Casas, D.d.l., Hanna, E.B., Bressand, F., et al.: Mixtral of experts. arXiv preprint arXiv:2401.04088 (2024)
15. Jin, P., Takanobu, R., Zhang, C., Cao, X., Yuan, L.: Chat-univi: Unified visual representation empowers large language models with image and video understanding (2023)
16. Kim, S., Kim, J.H., Lee, J., Seo, M.: Semi-parametric video-grounded text generation. arXiv preprint arXiv:2301.11507 (2023)

17. Lei, J., Li, L., Zhou, L., Gan, Z., Berg, T.L., Bansal, M., Liu, J.: Less is more: Clipbert for video-and-language learningvia sparse sampling. In: CVPR (2021)
18. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: ICML (2023)
19. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: ICML (2023)
20. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding (2023)
21. Li, Y., Chen, X., Hu, B., Zhang, M.: Llms meet long video: Advancing long video comprehension with an interactive visual adapter in llms (2024)
22. Lin, B., Zhu, B., Ye, Y., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. arXiv preprint arXiv:2311.10122 (2023)
23. Liu, H., Yan, W., Zaharia, M., Abbeel, P.: World model on million-length video and language with ringattention. arXiv preprint arXiv:2402.08268 (2024)
24. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
25. Liu, N.F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., Liang, P.: Lost in the middle: How language models use long contexts. Transactions of the Association for Computational Linguistics **12**, 157–173 (2024)
26. Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., et al.: Agentbench: Evaluating llms as agents. arXiv preprint arXiv:2308.03688 (2023)
27. Ma, C., Zhang, J., Zhu, Z., Yang, C., Yang, Y., Jin, Y., Lan, Z., Kong, L., He, J.: Agentboard: An analytical evaluation board of multi-turn llm agents. arXiv preprint arXiv:2401.13178 (2024)
28. Ma, F., Jin, X., Wang, H., Xian, Y., Feng, J., Yang, Y.: Vista-llama: Reliable video narrator via equal distance to visual tokens. arXiv preprint arXiv:2312.08870 (2023)
29. Mangalam, K., Akshulakov, R., Malik, J.: Egoschema: A diagnostic benchmark for very long-form video language understanding. arXiv preprint arXiv:2308.09126 (2023)
30. Momeni, L., Caron, M., Nagrani, A., Zisserman, A., Schmid, C.: Verbs in action: Improving verb understanding in video-language models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15579–15591 (2023)
31. Nguyen, E., Goel, K., Gu, A., Downs, G., Shah, P., Dao, T., Baccus, S., Ré, C.: S4nd: Modeling images and videos as multidimensional signals with state spaces. Advances in neural information processing systems **35**, 2846–2861 (2022)
32. OpenAI: Gpt-4 technical report (2023)
33. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems **35**, 27730–27744 (2022)
34. Pan, J., Lin, Z., Ge, Y., Zhu, X., Zhang, R., Wang, Y., Qiao, Y., Li, H.: Retrieving-to-answer: Zero-shot video question answering with frozen large language models (2023)
35. Papalampidi, P., Koppula, S., Pathak, S., Chiu, J., Heyward, J., Patraucean, V., Shen, J., Miech, A., Zisserman, A., Nematzdeh, A.: A simple recipe for contrastively pre-training video-first encoders beyond 16 frames. arXiv preprint arXiv:2312.07395 (2023)

36. Park, J.S., O'Brien, J., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative agents: Interactive simulacra of human behavior. In: Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. pp. 1–22 (2023)
37. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
38. Ren, S., Yao, L., Li, S., Sun, X., Hou, L.: Timechat: A time-sensitive multimodal large language model for long video understanding (2023)
39. Romero, D., Solorio, T.: Question-instructed visual descriptions for zero-shot video question answering (2024)
40. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools. Advances in Neural Information Processing Systems **36** (2024)
41. Shi, F., Chen, X., Misra, K., Scales, N., Dohan, D., Chi, E.H., Schärli, N., Zhou, D.: Large language models can be easily distracted by irrelevant context. In: International Conference on Machine Learning. pp. 31210–31227. PMLR (2023)
42. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S.: Reflexion: Language agents with verbal reinforcement learning. Advances in Neural Information Processing Systems **36** (2024)
43. Song, E., Chai, W., Wang, G., Zhang, Y., Zhou, H., Wu, F., Guo, X., Ye, T., Lu, Y., Hwang, J.N., et al.: Moviechat: From dense token to sparse memory for long video understanding. arXiv preprint arXiv:2307.16449 (2023)
44. Sun, Q., Wang, J., Yu, Q., Cui, Y., Zhang, F., Zhang, X., Wang, X.: Eva-clip-18b: Scaling clip to 18 billion parameters. arXiv preprint arXiv:2402.04252 (2024)
45. Sun, Y., Xue, H., Song, R., Liu, B., Yang, H., Fu, J.: Long-form video-language pre-training with multimodal temporal contrastive learning. Advances in neural information processing systems **35**, 38032–38045 (2022)
46. Surís, D., Menon, S., Vondrick, C.: Vipergpt: Visual inference via python execution for reasoning. Proceedings of IEEE International Conference on Computer Vision (ICCV) (2023)
47. Tay, Y., Dehghani, M., Abnar, S., Shen, Y., Bahri, D., Pham, P., Rao, J., Yang, L., Ruder, S., Metzler, D.: Long range arena: A benchmark for efficient transformers. arXiv preprint arXiv:2011.04006 (2020)
48. Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
49. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
50. Treisman, A.M., Gelade, G.: A feature-integration theory of attention. Cognitive psychology **12**(1), 97–136 (1980)
51. Wang, S., Zhao, Q., Do, M.Q., Agarwal, N., Lee, K., Sun, C.: Vamos: Versatile action models for video understanding (2023)
52. Wang, Y., Bertasius, G., Oh, T.H., Gupta, A., Hoai, M., Torresani, L.: Supervoxel attention graphs for long-range video modeling. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 155–166 (2021)

53. Wang, Y., Li, K., Li, Y., He, Y., Huang, B., Zhao, Z., Zhang, H., Xu, J., Liu, Y., Wang, Z., et al.: Internvideo: General video foundation models via generative and discriminative learning. arXiv preprint arXiv:2212.03191 (2022)
54. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems **35**, 24824–24837 (2022)
55. Wu, C.Y., Li, Y., Mangalam, K., Fan, H., Xiong, B., Malik, J., Feichtenhofer, C.: 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. pp. 13587–13597 (2022)
56. Xiao, J., Zhou, P., Yao, A., Li, Y., Hong, R., Yan, S., Chua, T.: Contrastive video question answering via video graph transformer. IEEE Transactions on Pattern Analysis; Machine Intelligence **45**(11), 13265–13280 (nov 2023)
57. Xiao, J., Shang, X., Yao, A., Chua, T.S.: Next-qa: Next phase of question-answering to explaining temporal actions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9777–9786 (2021)
58. Xu, J., Lan, C., Xie, W., Chen, X., Lu, Y.: Retrieval-based video language model for efficient long video question answering (2023)
59. Yang, A., Miech, A., Sivic, J., Laptev, I., Schmid, C.: Just ask: Learning to answer questions from millions of narrated videos. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1686–1697 (2021)
60. Yang, A., Miech, A., Sivic, J., Laptev, I., Schmid, C.: Zero-shot video question answering via frozen bidirectional language models. In: NeurIPS (2022)
61. Yang, J., Zhu, Y., Wang, Y., Yi, R., Zadeh, A., Morency, L.P.: What gives the answer away? question answering bias analysis on video qa datasets. arXiv preprint arXiv:2007.03626 (2020)
62. Yang, Z., Chen, G., Li, X., Wang, W., Yang, Y.: Doraemongpt: Toward understanding dynamic scenes with large language models. arXiv preprint arXiv:2401.08392 (2024)
63. Yao, S., Chen, H., Yang, J., Narasimhan, K.: Webshop: Towards scalable real-world web interaction with grounded language agents. Advances in Neural Information Processing Systems **35**, 20744–20757 (2022)
64. Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., Narasimhan, K.: Tree of thoughts: Deliberate problem solving with large language models. Advances in Neural Information Processing Systems **36** (2024)
65. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., Cao, Y.: React: Synergizing reasoning and acting in language models. arXiv preprint arXiv:2210.03629 (2022)
66. Ye, Q., Xu, G., Yan, M., Xu, H., Qian, Q., Zhang, J., Huang, F.: Hitea: Hierarchical temporal-aware video-language pre-training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15405–15416 (2023)
67. Yeung, S., Russakovsky, O., Mori, G., Fei-Fei, L.: End-to-end learning of action detection from frame glimpses in videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2678–2687 (2016)
68. Yu, S., Cho, J., Yadav, P., Bansal, M.: Self-chained image-language model for video localization and question answering. NeurIPS (2023)
69. Zhang, C., Lu, T., Islam, M.M., Wang, Z., Yu, S., Bansal, M., Bertasius, G.: A simple llm framework for long-range video question-answering. arXiv preprint arXiv:2312.17235 (2023)

70. Zhao, Y., Misra, I., Krähenbühl, P., Girdhar, R.: Learning video representations from large language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6586–6597 (2023)
71. Zheng, L., Yin, L., Xie, Z., Huang, J., Sun, C., Yu, C.H., Cao, S., Kozyrakis, C., Stoica, I., Gonzalez, J.E., Barrett, C., Sheng, Y.: Efficiently programming large language models using sglang (2023)
72. Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., et al.: Least-to-most prompting enables complex reasoning in large language models. In: ICLR (2023)