Too Many Frames, Not All Useful: Efficient Strategies for Long-Form Video QA

Jongwoo Park [1] **Kanchana Ranasinghe** [1] **Kumara Kahatapitiya** [1]
Wonjeong Ryoo **Donghyun Kim** [2] **Michael S. Ryoo** [1]
[1]Stony Brook University [2]Korea University
jongwopark@cs.stonybrook.edu

Abstract

Long-form videos that span across wide temporal intervals are highly information redundant and contain multiple distinct events or entities that are often loosely related. Therefore, when performing long-form video question answering (LVQA), all information necessary to generate a correct response can often be contained within a small subset of frames. Recent literature explore the use of large language models (LLMs) in LVQA benchmarks, achieving exceptional performance, while relying on vision language models (VLMs) to convert all visual content within videos into natural language. Such VLMs often independently caption a large number of frames uniformly sampled from long videos, which is not efficient and can mostly be redundant. Questioning these decision choices, we explore optimal strategies for key-frame selection that can significantly reduce these redundancies, namely *Hierarchical Keyframe Selector*. Our proposed framework, *LVNet*, achieves state-of-the-art performance at a comparable caption scale across three benchmark LVQA datasets: EgoSchema, NExT-QA, IntentQA. The code can be found at https://github.com/jongwoopark7978/LVNet

1 Introduction

Video understanding is a long-standing vision problem [1] with numerous real-world applications. It has been traditionally studied even before the era of differentiable representation learning, with hierarchical approaches focusing on longer videos [3, 15, 33, 13, 32]. Today, video understanding research involving the language modality is particularly popular, with tasks such as video question answering that involve generating human-style conversations in response to questions regarding videos [34, 51, 44].

The recent popularity of vision-language models (VLMs), particularly approaches connecting large language models (LLMs) to vision architectures [23, 21, 7], has resulted in significant improvements across visual question answering (VQA) tasks. These models demonstrate exceptional performance within the image domain, and their video variants [47, 28, 27] perform similarly on shorter videos, yet demonstrating limited performance on long-form video benchmarks [24, 16, 30, 31]. This can be attributed to the nature of long-form video benchmarks, which require both temporal sequence awareness and causal reasoning. An alternate line of works [52, 36, 16, 37] adapt LLMs that contain strong reasoning abilities for this task, using image VLMs to generate per-frame natural language descriptions, followed by video question answering purely within the language domain. However, these methods employ expensive VLMs to caption a large number of uniformly sampled frames. Such a design choice leading to high compute expense, is questioned in [4, 30, 37], and is the key

Workshop on Video-Language Models at 38th Conference on Neural Information Processing Systems (NeurIPS 2024).

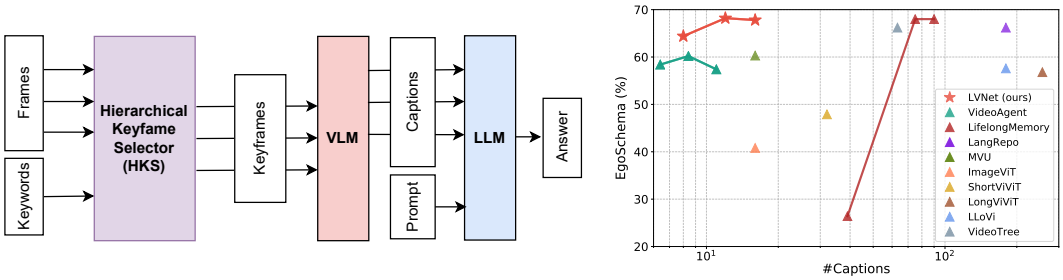

Figure 1: (left) Our proposed LVNet contains a novel Hierarchical Keyframe Selector module to select keyframes and followed by VLM and LLM for caption and answer generation. Hierarchical Keyframe Selector initially begins by processing dense frames and keywords with the lighter module and progressively exploits heavier and more performance-oriented modules on a small set of frames, resulting from the reduction of keyframe candidates, to ensure efficient computation. (right) LVNet achieves state-of-the-art performance on the EgoSchema subset while utilizing only a fraction of captioned frames. In particular, LVNet obtains its highest accuracy of 68.2% with 12 captions, outperforming VideoAgent, a model using a similar-scale captions, by 8%. A more detailed analysis of accuracy vs. the number of captions is provided in 4.3.

motivation for our exploration of *key frame selection*, i.e. identifying a minimal set of frames most useful for correctly answering a given video-question pair.

Therein, we propose LVNet, a framework containing a novel Hierarchical Keyframe Selector (HKS) that performs efficient key-frame selection followed by VLM and LLM for caption and answer generation as illustrated in Figure 1. Aligned with prior work [52, 39, 37], the per-frame captions are processed with a powerful LLM to generate correct answers for a given video-question pair. The scope of this work focuses on optimizing the prior two stages.

We summarize our key contributions as follows:

- Novel Hierarchical Keyframe Selector module which are composed of three submodules: Temporal Scene Clustering (TSC), Coarse Keyframe Detector (CKD), and Fine Keyframe Detector (FKD).
  - Temporal Scene Clustering performs non-uniform frame sampling by clustering visually similar frames, minimizing redundancy in long videos and capturing key scenes. The lightweight module is used to efficiently filter dense frames.
  - Coarse Keyframe Detector generates keywords, representing atomic activities, using the given query and LLM. It assigns confidence scores to frames based on keyword relevance, sub-sampling high-confidence frames for improved interpretability over visual-only selection.
  - Fine Keyframe Detector refines frame selection by combining multiple frames using visual templating and Visual Language Model (VLM), enabling higher-level reasoning and natural language output for improved accuracy over the CKD's keyword-based selection.
- Long-form Video Understanding framework requiring zero video-level training.

Our proposed LVNet achieves state-of-the-art results compared to the models that utilize a similar number of captions on three long-form video question answering benchmarks—EgoSchema[24], NExT-QA[41], and IntentQA[22] as described in 4.2. This demonstrates the strong performance and general applicability of our approach.

2 Related Work

Video Question Answering: Visual question answering (VQA), a relatively recent task, involves generating free-form and open-ended textual content conditioned on an image and natural language text [2]. Its video variant, Video-VQA [49] replaces images with videos. While multiple early datasets focus on querying objects or events based on referential and spatial relations [44, 51, 49], later tasks require explicit temporal understanding of sequential events [18, 19, 50]. More recent datasets focus on longer videos containing multiple actions and scenes spread over a wide time interval (termed long-form videos) [41, 20] with questions additionally requiring causal reasoning to

correctly answer. Referred to as long-form video question answering (LVQA), these benchmarks are constructed to specifically test strong causal and temporal reasoning [41] over long temporal windows [24]. Some works tackling such casual video VQA tasks leverage graph networks to model cross object / event relations [14, 42, 43]. A more recent line of works integrate large-language models (LLMs) to tackle this task [52, 36, 16, 37, 30, 39, 9] utilizing the strong reasoning skills of the LLMs. A common aspect is the use of a vision langauge model (VLM) to convert the frame level visual information into natural language. This is in turn input to the LLM which makes a final prediction.

Unlike these methods, our LVNet incorporates a unique Hierarchical Keyframe Selector that progressively reduces the number of keyframe candidates. Lighter modules are applied to dense frames, while heavier, more performance-focused modules are applied to a small set of frames for efficient computation. Additionally, LVNet does not require video-level training, unlike earlier supervised approaches.

Frame Selection in Videos: The task of frame selection in videos has been long explored in video [8, 53] with more recent works focused directly on long-form video question answering [4, 30]. Most similar to our work is [37] which employs an LLM based strategy for video frame selection. However, our proposed Hierarchical Keyframe Selector module differs with unique visual only operations as well as cross-modal operations handling much longer frame sequences extending up to 900 frames.

3 Method

In this section, we present our training-free (*i.e.* zero-shot) framework for long-form video QA, LVNet. Videos are a dense form of data with even a few seconds long clip being composed of 100s of frames (individual images). In the case of long-form videos, this frame count is even greater. However, the information necessary to answer a given question is often contained in a handful of those frames. Our framework tackles this challenge of selecting an optimal and minimal set of informative frames. We refer to this as keyframe selection. Given such a set of useful frames, we also establish optimal strategies for extracting their information using modern large language models (LLMs), taking into account their sequential nature.

Our proposed LVNet comprises of three components: a Hierarchical Keyframe Selector (HKS), a Vision Language Model (VLM), and a Large Language Model (LLM) as illustrated in Figure 1. The HKS, an efficient, hierarchical keyframe selector, is the core contribution of our work. First, the model processes 900 uniformly sampled frames and clusters them into distinct scenes using ResNet-18, a lightweight module that measures visual correspondence between frames. Next, it extracts keywords from a given natural language query via LLM and selects the frames most relevant to those keywords using CLIP-B/16. Finally, the selected frames are described in natural language by a more powerful and computationally intensive VLM. Finally, an LLM processes the language descriptions of the selected frames to answer a given query.

3.1 Background

Recent approaches utilizing LLMs for long video question answering (LVQA) [52, 36, 16, 30, 37] can be viewed as a composition of three sequential stages: a) frame selection, b) VLM based frame captioning, and c) LLM based answer generation. Note that the complexity of each stage varies across methods given their focus on different aspects of the LVQA task (*e.g.* frame selection in some is simply uniform sampling). In our work, we also follow this structure, but we focus on improving the frame selection stage. Under such a framework, our proposed HKS can serve as plug-in modules to replace the *frame selection* stage and the later two stages are similar to these prior works.

3.2 Architecture

Consider a long video, $\mathbf{x} \in \mathbb{R}^{T \times C \times H \times W}$ with T, C, H, W for frames, channels, height, and width respectively. Its paired natural language query is referred as $\mathbf{q}$. Also consider a frame in $\mathbf{x}$ at timestamp t as $\mathbf{x}[t] \in \mathbb{R}^{C \times H \times W}$. Our goal is to output the suitable response to this query based on information contained in the video. We refer this correct response as $\mathbf{r}$.

Our LVNet processes a given video-query $(\mathbf{x}, \mathbf{q})$ pair to output a response, $\hat{\mathbf{r}}$. The HKS module initially processes this video-query pair, selects T' keyframes, and outputs a deterministically sub-

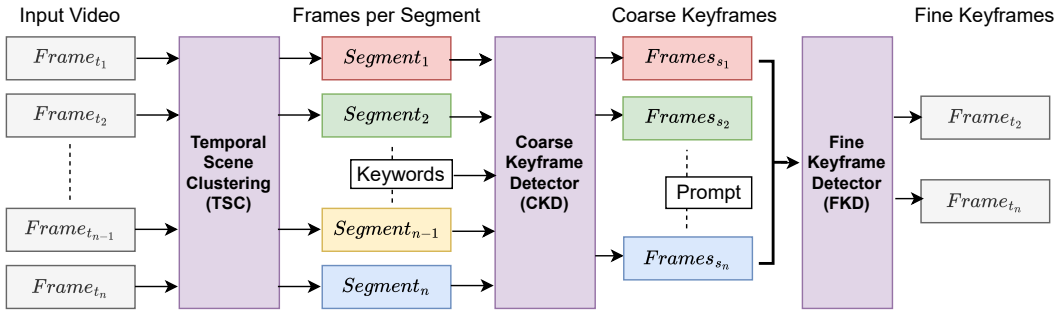

Figure 2: **Detailed architecture of Hierarchical Keyframe Selector:** Input frames first go through a Temporal Scene Clustering (TSC) mechanism, which divides the long-video into scenes, enabling per-scene subsampling. Next, a Coarse Keyframe Detector (CKD) is applied to select frames best-aligned with keywords relevant to the query. Finally, a Fine Keyframe detector further subsamples frames, by refining keyword alignements through a templated visual prompting.

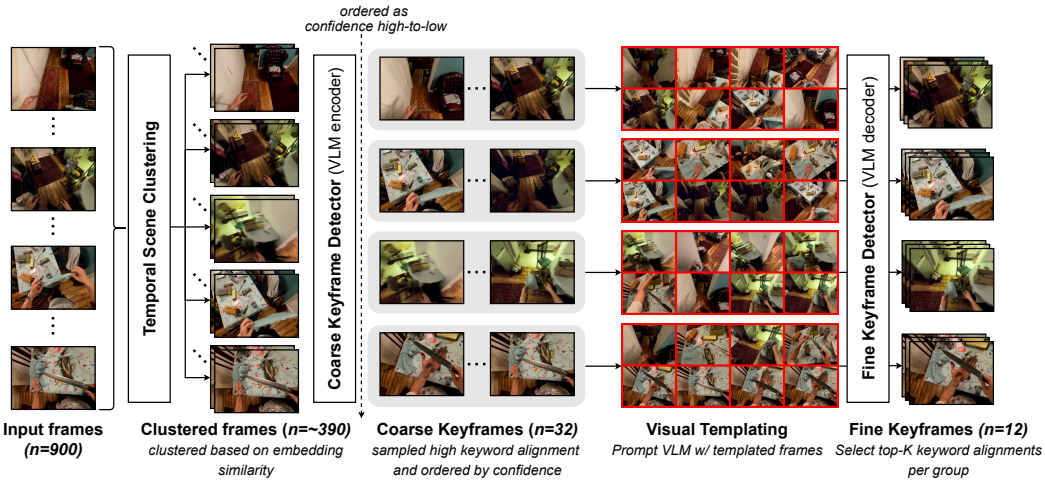

Figure 3: **Qualitative example**: We illustrate a challenging long-video QA scenario from EgoSchema [24]. We consider an input of 900 frames, which first get clustered into scenes and subsampled to retain around 390 frames. Next, the Coarse Keyframe Detector selects only 32 frames out of them, based on the alignment with keywords (Here, keywords are extracted based on answer options, via an LLM). Such coarse keyframes are then ranked based on the combination of confidence value and temporal span, and grouped into four sets, each containing eight frames. These sets are then processed through visual templating (*i.e.* simple concatenation across space) and fed into a VLM for Fine Keyframe Detection, resulting in just 12 frames.

sampled video $\mathbf{x}' \in \mathbb{R}^{T' \times C \times H \times W}$. Each of these T' frames is then passed through the captioning stage of our VLM to generate a set of natural language descriptions, $D = \{d_1, d_2, ...d_{T'}\}$ where d_i describes the frame $\mathbf{x}'[i]$. Finally, the LLM processes all descriptions D and the query $\mathbf{q}$ to generate response $\hat{\mathbf{r}}$.

Our unique contributions here are the Hierarchical Keyframe Selector (HKS) and optimal prompts and description templates for the LLM when generating the final response which is also important as suggested by our experiments. The overall architecture is given in Figure 1 (left).

3.3 Hierarchical Keyframe Selector

We now describe our proposed Hierarchical Keyframe Selector (HKS) module. As illustrated in Figure 2, our HKS comprises of three submodules, namely 1) Temporal Scene Clustering (TSC), 2) Coarse Keyframe Detector (CKD), and 3) Fine Keyframe Detector (FKD). The inputs to HKS

contain T frames with each of the three stages TSC, CKD, and FKD reducing this frame count to T_a, T_b, and $T_c = T'$ respectively.

Temporal Scene Clustering (TSC): The role of TSC is to perform visual content aware preliminary frame sampling. The established approach for preliminary frame selection is uniform sampling. In contrast, TSC extracts per-frame visual features using a deep neural network followed by an iterative clustering procedure to identify n non-overlapping frame sets. Within each of the n sets, we uniformly sample $\leq \tau$ frames obtaining a total of $T_a \leq \tau \times n$. Our iterative clustering procedure is outlined in Algorithm 1. It calculates pairwise distances between all frames accounting for intra-frame local information using the extracted per-frame features, followed by n iterative frame similarity based grouping operations.

Our reasoning for TSC is that videos, especially long-form ones, contain extensive information redundancy across its temporal dimension. For example, in a long sub-sequence of mostly static frames from a single scene, beyond its first frame the rest may contain little additional information. We hypothesize that our TSC module will utilize frame level feature similarity to cluster the frames, such that each cluster corresponds to a common scene or motion. A single cluster could contain just one frame or significantly more based on frame feature similarities. This leads to a *non-uniform sampling of frames* across the entire video. In contrast to uniform sampling across the entire videos where each timestamp is treated equally, in our formulation the uniform sampling within segments will allow more frames to be sampled from information heavy temporal regions.

Coarse Keyframe Detector (CKD): Unlike the TSC in prior stage, CKD reasons across both visual and language modalities (using the paired textual query) to further sub-sample the T_a frames output from the TSC module. The inputs to this CKD stage are the T_a frames and the paired query $\mathbf{q}$. The module generates a set of keywords K_{CKD}, selects T_b frames conditioned on these keywords, and outputs the selected T_b frames with a confidence value and keyword assigned to each frame.

The CKD contains three elements: a keyword generation strategy, a dual-encoder image-text model, and an iterative algorithm for similarity based confidence assignment. The keyword generation strategy utilizes the given query, $\mathbf{q}$ alongside a hand-crafted templating operation or an LLM to select or generate suitable keywords. The dual-encoder image-text model uses a spatially aware contrastive langauge image pre-training (CLIP) network from [29]. For confidence assignment, we construct an algorithm as outlined in Algorithm 2 which processes two lists, one of frames and one of keywords, and then calculates their pairwise likelihood of occurrence to assign each frame a confidence value (that reflects its usefulness to answer the query, $\mathbf{q}$).

The keyword generation strategy converts the query $\mathbf{q}$ into a set of keywords, K_{CKD} where the number of keywords $|K_{\text{CKD}}| \in [8, 25]$. The dual-encoder model converts the keyword set K_{CKD} and the frame set (containing T_a frames) into two sets of embeddings. The embeddings of each set are comparable to each other using a cosine similarity distance metric given the training formulation of the CLIP model. These two embedding sets are processed by our confidence assignment algorithm in Appendix A.2 which assigns each of the T_a frames a confidence score and a keyword. Note that the same keyword can be assigned to multiple frames. However, given the much higher frame count compared to the keyword count and the cross-frame redundancy, we do not assign multiple keywords to a single frame.

The key role of this module is language conditioned frame selection. For a single query, there can be multiple regions in a video that are highly informative but not useful or relevant in answering that query. A single query can also contain multiple different concepts and attributes that must be given attention to construct a correct answer: the keyword generation attempts to capture each of these distinct attributes. On the visual modality, a single frame will also encode multiple concepts and attributes. Our design choice for the spatially aware CLIPpy dual-encoder VLM from [29] is motivated by this nature of individual frames. Finally, confidence assignment takes into account these multiple modes of information within each frame and the query and suitably assigns a confidence score to each frame that reflects its relevance to the query. We also highlight how the confidence scores are directly linked to the related keyword (i.e. reason that makes the frame relevant), leading to better interpretability and the ability to perform further keyword-based refinement in later stages.

Fine Keyframe Detector (FKD): In the prior CKD stage, cross-modal selection utilizes a dual-encoder VLM that is constrained by the set of keywords provided and performs limited reasoning at frame level. In contrast, FKD uses a *visual templating module* to combine multiple frames and uses

VLM to generate open-ended natural language output through higher-level reasoning. The input in this stage is the set of F_b frames, with each frame having an assigned confidence score and keyword.

Our visual templating module partitions the T_b frames into sets of 8 ordered by their confidence scores, arranges frame sets as grids to form a collage-style image, and annotates that image with visually identifiable tags corresponding to each frame. We further illustrate this process in Figure 3 (see Visual Templating column). Each of these visual templated images also contain a subset of the keywords that correspond to their 8 images. These resulting visual templated images along with a prompt containing their associated keywords and instructions to select a frame subset based on valid association between keywords and images (see Appendix A.3 for details) are input to the VLM. The output of the VLM is used to select a subset of each 8 image group. These frames are collected as the output of the FKD stage, overall resulting in T_c frames.

The purpose of the initial visual templating module is to allow reasoning across a set of frames using the image-text VLM (which is trained to process a single image at time). This partitioning of the input T_b frames is performed based on confidence scores from the prior stage and timestamps. The eight frames with top confidence scores are grouped into the first visual template, followed by the next eight and so forth. This ensures the VLM selects both high confidence concepts and low confidence concepts, accounting for biases and weaknesses in our CKD stage. After that, we temporally reorder some image sets with low confidence scores to cover keyframes distributed across long-range segments, while the sets with high confidence scores concentrate on keyframes in short-range segments. A total of 16 low-score frames are temporally reordered in this process. The algorithm is described in Algorithm 3 and the prompting technique is explained in Appendix A.3. Our intuition is that such a mechanism allows one to best utilize the complementary strengths of two different VLMs from CKD and FKD stages for better frame selection overall.

3.4 Comparison with Other Keyframe Selection Methods:

We aim to highlight the main advantage of the Hierarchical Keyframe Selector over other existing keyframe selection methods. Models like VideoAgent and TraveLER provide useful comparisons, as they work with a similar-scale number of captions. VideoAgent and TraveLER rely on uniform frame selection in the first iteration without analyzing the entire video. They identify important segments based solely on these initial frames and the LLM's response, which can be problematic if the initial uniformly selected frames are not representative of the entire video or if the LLM misinterprets the captions and prompts. In such cases, the LLM might incorrectly identify segments for further analysis. If the LLM fails to pinpoint the correct segment initially, the entire process can break down because subsequent frames will be similar to the first set, leading the LLM to continuously select frames within or near the initial segment. Additionally, for videos that are as challenging or more difficult than EgoSchema in terms of temporal complexity and activities, these models may require numerous iterations to finalize keyframes selection. This results in higher computational and latency costs, as it necessitates numerous runs of resource-intensive VLM and LLM models.

In contrast, our method analyzes the entire video with high frame rates using a lightweight ResNet-18 [12] and segments the video non-uniformly based on scene continuity. We then select several frames in each segment by measuring feature similarity between frame features and keywords using the CLIP-B/16 (0.12B) [29] which is lighted than VideoAgent's EVA-CLIP-8Bplus (8B). By reviewing the entire video and non-uniformly selecting keyframes based on scene continuity and similarity scores, these keyframes accurately represent the question-based important frames distribution in the entire video. Furthermore, we use VLM for a fine-grained selection of keyframes, improving keyframe selection when CLIP-B/16 struggles to understand detailed atomic activities in the frames. By hierarchically segmenting the video with different modules, the resulting segments and keyframes are more reliable than those from VideoAgent. Even with more challenging videos, our process only needs to go through the video once to collect keyframes, maintaining computational efficiency.

4 Experiments

In this section, we first discuss our experimental setup followed by quantitative evaluations comparing to existing baselines and ablations of our proposed components. We then present qualitative results for our method and outline some limitations of our approach.

4.1 Experimental Setup

Datasets: Given the training free nature of our framework, we do not utilize any video datasets for training. Datasets are used purely for evaluation. We select three benchmark video visual question answering datasets focused on long-form videos for this purpose: EgoSchema [24], NExT-QA [41], and IntentQA [22]. The first dataset, EgoSchema, is a very long-form video question-answering dataset with 5031 questions. For each question, EgoSchema requires the correct answer to be selected between five given options based on a three-minute-long video clip. The second dataset, NExT-QA, is another rigorously designed video question answering benchmark containing questions that require causal & temporal action reasoning, and common scene comprehension to correctly answer. These questions are further classified as Causal (Cau.), Temporal (Tem.), and Descriptive (Des.) and we evaluate on its validation set containing 4996 questions over 570 videos. The third dataset, IntentQA [22] is based on NExT-QA videos corresponding to temporal and causal reasoning quetions. It consists of 16k multiple-choice questions which are classified as Why?, How? or Before/After (B./A.).

Model Choices: For the HKS module, we use the ResNet-18 [12] for the TSC, CLIP-B/16 [29] for the CKD and GPT-4o for the FKD. We select ResNet-18 and CLIP-B/16 due to their smaller models sizes—0.01B and 0.12B, respectively–which are significantly lighter compared to GPT-4o, whose model size is expected to be on the scale of 100B-1T. This makes them well-suited for filtering dense frames efficiently. In line with previous state-of-the-art work [39, 52, 36], we employ GPT API, especially GPT-4o, for both VLM and LLM. This choice is driven by its cost-effectiveness and lighter computational requirements compared to GPT-4. GPT-4o is used as the VLM for generating captions and as the LLM for answering questions in our framework.

Table 1: Combined Results on EgoSchema [24], NExT-QA [41], and IntentQA [22]. We compare LVNet against prior zero-shot models across three datasets, highlighting different task splits. The models are ordered based on the number of captions processed before answering the question. LVNet achieves state-of-the-art accuracies of 71.7%, 61.1%, and 72.9% on the three datasets, respectively, using just 12 frames compared to the models using the similar number of captions. Models with video-caption pretraining or utilizing significantly more captions than the 12 frames used by LVNet are de-emphasized in grey or downplayed in light green to ensure fairness with image-level pretraining or highlight caption efficiency.

Model	EgoSchema		NExT-QA					IntentQA				
	Cap.	Acc. (%)	Cap.	Cau. (%)	Tem. (%)	Des. (%)	All (%)	Cap.	Why? (%)	How? (%)	B./A. (%)	All (%)
Vamos [36]	-	48.3	-	-	-	-	-	-	-	-	-	-
IG-VLM [17]	-	59.8	-	69.8	63.6	74.7	68.6	-	-	-	-	65.3
VideoLLaMA 2 [5]	-	53.3	-	-	-	-	-	-	-	-	-	-
InternVideo2 [38]	-	60.2	-	-	-	-	-	-	-	-	-	-
Tarsier [35]	-	61.7	-	-	-	-	79.2	-	-	-	-	-
VIOLET [10]	5	19.9	-	-	-	-	-	-	-	-	-	-
mPLUG-Owl [46]	5	31.1	-	-	-	-	-	-	-	-	-	-
VideoAgent [37]	8.4	54.1	8.2	72.7	64.5	81.1	71.3	-	-	-	-	-
MVU [30]	16	37.6	16	55.4	48.1	64.1	55.2	-	-	-	-	-
MoReVQA [25]	30	51.7	16	70.2	64.6	-	69.2	-	-	-	-	-
TraveLER [37]	-	-	∼25	70.0	60.5	78.2	68.2	-	-	-	-	-
VFC [26]	-	-	32	45.4	51.6	64.1	51.5	-	-	-	-	-
SeViLA† [48]	32	22.7	32	61.3	61.5	75.6	63.6	32	-	-	-	60.9
ProViQ [6]	60	57.1	60	-	-	-	64.6	-	-	-	-	-
VideoTree [40]	63.2	61.1	≤56	75.2	67.0	81.3	73.5	≤56	-	-	-	66.9
FrozenBiLM [45]	90	26.9	-	-	-	-	-	-	-	-	-	-
LifelongMemory [39]	90	62.1	-	-	-	-	-	-	-	-	-	-
LangRepo [16]	180	41.2	90	64.4	51.4	69.1	60.9	90	62.8	62.4	47.8	59.1
LLoVi [52]	180	50.3	90	69.5	61.0	75.6	67.7	90	68.4	67.4	51.1	64.0
LVNet (ours)	12	61.1	12	75.0	65.5	81.5	72.9	12	75.0	74.4	62.1	71.7

4.2 Evaluation

Quantitative Results: We evaluate LVNet on the EgoSchema, NExT-QA, and IntentQA dataset and present our results in Table 1. Models with video-caption pretraining are de-emphasized in grey to ensure fairness with image-level pertaining. Models utilizing significantly more captions than the 12 frames are downplayed in light green to consider caption efficiency. For EgoSchema, we achieve 61.1% on the fullest, the highest among the models utilizing approximately 12 captions. This result outperforms VideoAgent, the next best model using 8.4 captions, by 7%. To ensure a fair comparison, we ablate the number of captions for LVNet in 2c, observing that LVNet with 8 captions

Table 2: **Ablations on EgoSchema [24]:** We evaluate different design decisions of our framework on EgoSchema 500-video subset for zero-shot video VQA.

(a) **Visual Templating Order**: Instead of default confidence-based ordering for Visual Templating, we consider Temporal ordering.

Visual Templating order	Acc. (%)
Temporal	65.2
Hybrid (Confidence + Temporal)	68.2

(b) **Effect of Hierarchical Keyframe Modules**: We measure the accuracy of the LVNet by adding hierarchical keyframe modules progressively.

TSC	CKD	FKD	Acc.
✓	✓	✓	68.2
✓	✓	✗	65.8
✓	✗	✗	64.5
✗	✗	✗	62.6

(c) **Number of Frame Captions**: We ablate different number of frames considered for captioning in our framework. Compared to VideoAgent [37], ours is more stable with consistently better performance.

Frames	6.4	8	8.4	11	12	16
VideoAgent[37]	58.4	-	60.2	57.4	-	-
LVNet (ours)	-	64.4	-	-	68.2	67.8

still outperforms VideoAgent with 8.4 captions by 4.2% in the EgoSchema subset. The superior accuracy of LVNet over other keyframe selectors, such as those used in VideoAgent and TraveLER, is discussed in Section 3.4.

We next evaluate on the NExT-QA[41] dataset. This dataset has a particular focus on both temporal and casual reasoning based question-answer pairs. Our approach achives state-of-the-art performance on this benchmark outperforming prior work among the models utilizing approximately 12 captions. In fact, our LVNet with just 12 frame captions achieves 72.9% overall accuracy, outperforming VideoAgent [37] (71.3% at 8.2 captions) by 1.6%.

We finally evaluate on the IntentQA[22] dataset. Our approach achieves state-of-the-art results on this benchmark, outperforming all prior work, including the de-emphasized models with video-caption pretraining and the downplayed models utilizing significantly more captions than 12 frames. In fact, LVNet shows a substantial improvement (+4.8%) over the next best model, VideoTree [40] (66.9% with 90 captions), achieving 71.7% accuracy with only 12 frame captions.

Given the generative nature of VQA tasks as well as the limited availability and noisy nature of fully-annotated video VQA corpora, building generalizable fully-supervised models are challenging for these tasks. Nevertheless, we highlight how our zero-shot and video level training-free framework is competitive with the best supervised approaches on this dataset. This indicates the promise of utilizing pretrained models, especially those equipped with extensive world knowledge and reasoning skills from alternate modality specific learning (i.e. in our cases image domain VLMs and language domain LLMs).

Qualitative Analysis of the Hierarhical Keyframe Selector: We compare the open-ended responses of LVNet and the uniform sampling method in Figure 4 to understand the effectiveness of the hierarchical keyframe selector in LVNet. The frames chosen by LVNet and the naive uniform sampling method are indicated by blue and red checkmarks in the images, respectively. LVNet selects frames at 5, 69, and 135 seconds by executing the hierarchical keyframe selector and generates captions based on those frames. When we feed the concatenated captions to the LLM to answer the given question: *"Based on the video, what are the three main types of tools that C uses..."* in an open-ended manner, the output identifies two main activities: welding torches and measuring tapes, among the three main activities described in Option 3 (welding handle, hammer, and measuring tape), which is the correct answer. This leads LVNet to choose the correct option.

In contrast, the uniform sampling method selects frames at 0, 16, and 32 seconds and generates captions based on those frames. Similarly, when we feed the concatenated captions to the LLM to answer the same question, the output identifies only one activity—welding tools—resulting in the selection of the incorrect option. This example highlights the importance of keyframe selection and demonstrates the effectiveness of hierarchical keyframe selection in LVNet.

4.3 Ablations

In this section, we present ablations on key design decisions such as the sorting order in FKD, the number of frames for captions, and the effect of different components in HKS. In all ablations, we

Time(s):	0	5	16	32	69	135
Caption:	A person is using a hacksaw…	… holding a welding torch …	A person is welding or soldering …	C is using a power tool …	A person is working on unwrapping…	…They are manipulating a measuring tape …

Prompt: {Concatenated Captions from ✓ or ✓ }. I request you to answer the following question based on the preceding descriptions in less than 50 words. Question: Based on the video, what are the three main types of tools that c uses, and how do their roles in shaping the iron differ from one another?

LLM Output:

✓ **LVNet (Ours)**: Based on the video, the three main types of tools used are welding torches, pliers, and measuring tapes…

✓ **Uniform Sampling**: The three main types of tools used are hacksaws, welding tools, and power tools…

GT (option 3): The three main types of tools that c uses are a welding handle, a hammer, and a measuring tape…

Figure 4: **Open-ended Responses from LVNet vs Uniform Sampling**: The frames chosen by LVNet and the naive uniform sampling method are indicated with blue and red checkmarks, respectively. LVNet identifies both welding torches and measuring tapes, choosing the correct option, whereas uniform sampling only detects welding tools and selects the incorrect answer. The blue, red, and purple highlights correspond to the three main activities in the video—welding a handle, using a hammer, and using a measuring tape, respectively.

use a subset of EgoSchema [24], composed of 500 videos. Additional ablations about *Choice of LLM* and *Effect of Patch Size on Keyword Matching in CKD* are in Appendix A.1

Visual Templating Order: In visual templating, prioritizing frames by keyword confidence scores followed by reordering low-confidence frames based on timestamp is more effective than relying solely on temporal order. In this hybrid approach, high-confidence frames are selected from shorter segments by sampling three keyframes per set of eight, grouped by confidence rather than timespan. Conversely, low-confidence keyframes, which are crucial but more visually challenging for keyword matching, are sampled from broader video sections, with three frames per set of eight grouped by timestamp. This hybrid approach outperforms purely temporal ordering by 3%.

Number of Frame Captions: We conducted an ablation study on the number of frame captions used in our setup, comparing it to VideoAgent [37], which operates with a similarly small number of captions. Our findings show that LVNet achives its highest accuracy 68.2% with 12 captions, while even its lowest accuracy 64.4% with 8 captions still surpasses VideoAgent's best accuracy 60.2% with 8.4 captions.

Effect of Hierarchical Keyframe Modules: This table demonstrates the impact of incrementally adding the temporal scene clustering (TSC), coarse keyframe detector (CKD), and fine keyframe detector (FKD) modules. Without any of these modules, the model relies on uniform sampling and achieves 62.6%. When TSC is added and 12 frames are selected uniformly, the accuracy increases to 64.5%. Adding both TSC and CKD raises the accuracy to 65.8%. Finally, incorporating all three modules—TSC, CKD, and FKD—into the model, which is LVNet, results in an accuracy of 68.2%. This demonstrates the importance of including all modules in LVNet for optimal performance.

5 Conclusion

We proposed a novel approach for Long-form Video Question Answering (LVQA) that achieves state-of-the-art performance compared to the model using the similar-scale captions across 3 benchmarks datasets. Our Hierarchical Keyframe Selector demonstrates the effectiveness of keyframe selection in understanding a very long-form video QA. Additionally, we highlight the zero-shot capability for long-form video comprehension of our LVNet framework, which requires no video-level training. Our experiments showcase its significant advantage over previous methods.

Acknowledgements: This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government [24ZB1200, Research of Human-centered autonomous intelligence system original technology]. This work was also supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. RS-2024-00336738, Development of Complex Task Planning Technologies for Autonomous Agents, 100%)

References

[1] Jake K. Aggarwal and Michael S. Ryoo. Human activity analysis. *ACM Computing Surveys (CSUR)*, 43:1 – 43, 2011.

[2] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Vqa: Visual question answering. *International Journal of Computer Vision*, 123:4 – 31, 2015.

[3] James F Allen and George Ferguson. Actions and events in interval temporal logic. *Journal of logic and computation*, 4(5):531–579, 1994.

[4] S. Buch, Cristobal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles. Revisiting the "video" in video-language understanding. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2907–2917, 2022.

[5] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024.

[6] Rohan Choudhury, Koichiro Niinuma, Kris M Kitani, and László A Jeni. Zero-shot video question answering with procedural programs. *arXiv preprint arXiv:2312.00937*, 2023.

[7] 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.

[8] James Davis and Aaron Bobick. The representation and recognition of action using temporal templates. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2736–2744, 1997.

[9] Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. *arXiv preprint arXiv:2403.11481*, 2024.

[10] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. An empirical study of end-to-end video-language transformers with masked visual modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22898–22909, 2023.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] Somboon Hongeng, Ram Nevatia, and Francois Bremond. Video-based event recognition: activity representation and probabilistic recognition methods. *Computer Vision and Image Understanding*, 96(2): 129–162, 2004.

[14] Pedram Hosseini, David A. Broniatowski, and Mona Diab. Knowledge-augmented language models for cause-effect relation classification. In *Proceedings of the First Workshop on Commonsense Representation and Reasoning (CSRR 2022)*, pages 43–48, Dublin, Ireland, 2022. Association for Computational Linguistics.

[15] Yuri A. Ivanov and Aaron F. Bobick. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):852–872, 2000.

[16] Kumara Kahatapitiya, Kanchana Ranasinghe, Jongwoo Park, and Michael S Ryoo. Language repository for long video understanding, 2024.

[17] Wonkyun Kim, Changin Choi, Wonseok Lee, and Wonjong Rhee. An image grid can be worth a video: Zero-shot video question answering using a vlm. *arXiv preprint arXiv:2403.18406*, 2024.

[18] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. TVQA: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[19] Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. TVQA+: Spatio-temporal grounding for video question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8211–8225, Online, 2020. Association for Computational Linguistics.

[20] Jiangtong Li, Li Niu, and Liqing Zhang. From representation to reasoning: Towards both evidence and commonsense reasoning for video question-answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[21] 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.

[22] Jiapeng Li, Ping Wei, Wenjuan Han, and Lifeng Fan. Intentqa: Context-aware video intent reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11963–11974, 2023.

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

[24] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *ArXiv*, abs/2308.09126, 2023.

[25] Juhong Min, Shyamal Buch, Arsha Nagrani, Minsu Cho, and Cordelia Schmid. Morevqa: Exploring modular reasoning models for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13235–13245, 2024.

[26] Liliane Momeni, Mathilde Caron, Arsha Nagrani, Andrew Zisserman, and Cordelia Schmid. Verbs in action: Improving verb understanding in video-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15579–15591, 2023.

[27] Salman Khan Muhammad Maaz, Hanoona Rasheed and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *ArXiv 2306.05424*, 2023.

[28] Pinelopi Papalampidi, Skanda Koppula, Shreya Pathak, Justin Chiu, Joe Heyward, Viorica Patraucean, Jiajun Shen, Antoine Miech, Andrew Zisserman, and Aida Nematzdeh. A simple recipe for contrastively pre-training video-first encoders beyond 16 frames. *arXiv preprint arXiv:2312.07395*, 2023.

[29] Kanchana Ranasinghe, Brandon McKinzie, Sachin Ravi, Yinfei Yang, Alexander Toshev, and Jonathon Shlens. Perceptual grouping in contrastive vision-language models. In *ICCV*, 2023.

[30] Kanchana Ranasinghe, Xiang Li, Kumara Kahatapitiya, and Michael Ryoo. Understanding long videos in one multimodal language model pass, 2024.

[31] Ruchit Rawal, Khalid Saifullah, Ronen Basri, David Jacobs, Gowthami Somepalli, and Tom Goldstein. Cinepile: A long video question answering dataset and benchmark. *arXiv preprint arXiv:2405.08813*, 2024.

[32] Michael S. Ryoo and Jake K. Aggarwal. Recognition of composite human activities through context-free grammar based representation. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2:1709–1718, 2006.

[33] Yifan Shi, Yan Huang, David Minnen, Aaron Bobick, and Irfan Essa. Propagation networks for recognition of partially ordered sequential action. In *CVPR*, 2004.

[34] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[35] Jiawei Wang, Liping Yuan, and Yuchen Zhang. Tarsier: Recipes for training and evaluating large video description models. *arXiv preprint arXiv:2407.00634*, 2024.

[36] Shijie Wang, Qi Zhao, Minh Quan Do, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. Vamos: Versatile action models for video understanding. *arXiv preprint arXiv:2311.13627*, 2023.

[37] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent, 2024.

[38] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *arXiv preprint arXiv:2403.15377*, 2024.

[39] Ying Wang, Yanlai Yang, and Mengye Ren. Lifelongmemory: Leveraging llms for answering queries in long-form egocentric videos, 2024.

[40] Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. *arXiv preprint arXiv:2405.19209*, 2024.

[41] 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 (CVPR)*, 2021.

[42] Junbin Xiao, Angela Yao, Zhiyuan Liu, Yicong Li, Wei Ji, and Tat-Seng Chua. Video as conditional graph hierarchy for multi-granular question answering. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2804–2812, 2022.

[43] Junbin Xiao, Pan Zhou, Tat-Seng Chua, and Shuicheng Yan. Video graph transformer for video question answering. In *European Conference on Computer Vision*, pages 39–58. Springer, 2022.

[44] 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 *ACM Multimedia*, 2017.

[45] 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.

[46] 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.

[47] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. *ArXiv*, abs/2305.06988, 2023.

[48] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. *Advances in Neural Information Processing Systems*, 36, 2024.

[49] Zhou Yu, D. Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. *ArXiv*, abs/1906.02467, 2019.

[50] 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 (AAAI)*, 2019.

[51] Kuo-Hao Zeng, Tseng-Hung Chen, Ching-Yao Chuang, Yuan-Hong Liao, Juan Carlos Niebles, and Min Sun. Leveraging video descriptions to learn video question answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), 2017.

[52] Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering. *arXiv preprint arXiv:2312.17235*, 2023.

[53] Zhichen Zhao, Huimin Ma, and Shaodi You. Single image action recognition using semantic body part actions. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017.