Recurrent Attention-based Token Selection for Efficient Streaming Video-LLMs

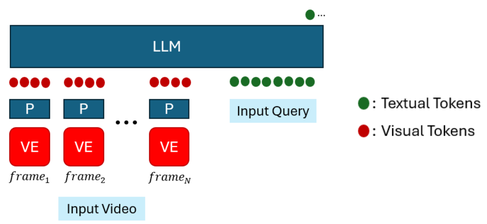
Vaggelis Dorovatas, Soroush Seifi, Gunshi Gupta, Rahaf Aljundi

NEURAL INFORMATION PROCESSING SYSTEMS

In this work, we tackle the problem of online long video understanding with video-LLMs

Video-LLMs

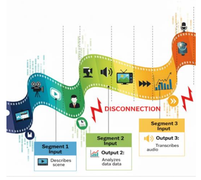
- Generate a textual response given a video & a textual query:



- **Typically trained in short videos** (seconds or a few mins).

Online Long Video Understanding

- Long Stream is **continually received** (online)
- **Input queries** can arrive **at any time**
- When the received frames exceed the context limit, we have "**in-context disconnection**".



Previous Streaming Video Understanding Works

Independent Clip Processing

- Split the long stream into independent segments
- Generate captions for each and use them for answering queries
  - ↪ **Lacks continuity**

Visual Memory

- Compression module to compress visual tokens
- Usually requires **training**
  - Coupled with the base video-LLM
- **Doesn't exploit the LLM's reasoning abilities** for compression

Storing the full KV cache

- High **memory requirements**
- **Slow retrieval**
  - Considers all previous KVs
- **High redundancy** may confuse retrieval

Our Approach: *rLiVS*

"informed"

- **Training-free** (model-agnostic)
- **Efficient Token Selection based on Model's attention** (we don't store everything)
- Continuous video stream → **group frames into segments**
- **Recurrency** for continuity over segments
  - ↪ ***Full context = History Tokens + Current segment's visual tokens***
  - ↪ **History Tokens = Memory FIFO** queue that stores selected tokens
- Generate captions for each segment (**LLM thoughts**) and save them in the long-term memory M.

Online Video Processing in detail

Given current short clip X_V^i and history tokens S^i, we have:

- **Caption Generation (LLM thought)**

$$C^i = videoLLM(X_I, X_V^i, S^i)$$

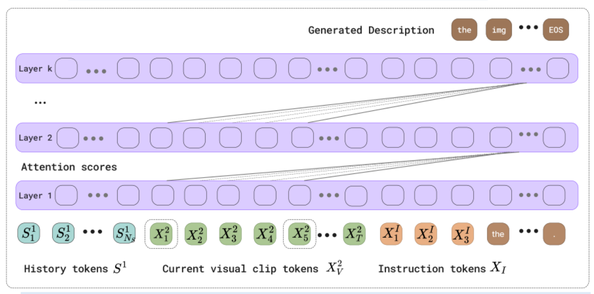
- **Token Selection & Memory Update**

$$\hat{X}_V^i = attn\_selection(X_V^i, C^i)$$
$$S^{i+1} = update(S^i, \hat{X}_V^i)$$

- **Storing LLM thought:** $M.append(C^i)$

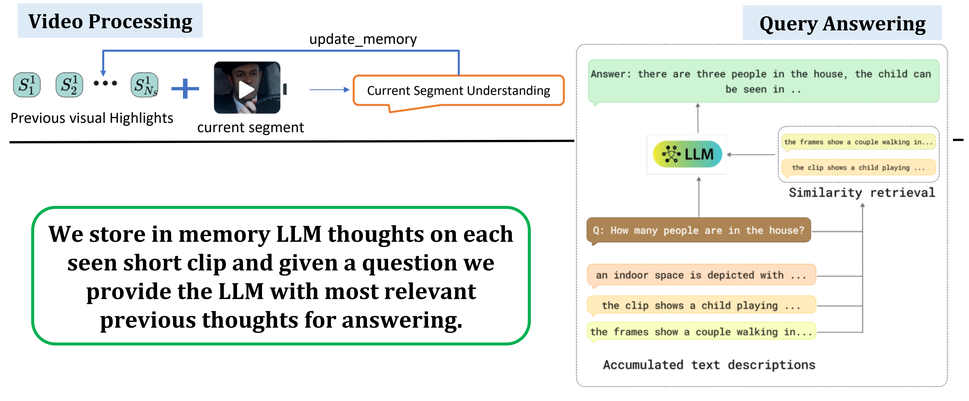
*We note that $S^0 = []$ and M are initially empty.

Incoming frames are grouped into short clips



Full model context is split into *Memory + Current visual tokens*

rLiVS at inference



We store in memory LLM thoughts on each seen short clip and given a question we provide the LLM with most relevant previous thoughts for answering.

Streaming VQA

Method	RVS-Ego		RVS-Movie		Latency	VRAM	KV-Cache
	Acc.	Sco.	Acc.	Sco.			
MovieChat [28]	50.7	3.4	36.0	2.3	-	-	-
LLaMA-VID [21]	53.4	3.9	48.6	3.3	-	-	-
Flash-VStream-7B [38]	57.3	4.0	53.1	3.3	2.1s	19GB	-
VideoScan [20]	60.9	4.0	54.1	3.5	2.1s	18GB	-
LLaVA-OV 0.5B							
↪ ReKV [9]	54.7	3.7	44.6	3.4	1.6s	19GB	4.0 GB/h
↪ **rLiVS (Ours)**	57.6	3.8	51.3	3.4	1.5s	11GB	-
LLaVA-OV 7B							
↪ ReKV [9]	63.7	4.0	54.4	3.6	2.7s	36GB	18.8 GB/h
↪ **rLiVS (Ours)**	65.3	4.0	**57.7**	3.6	**1.9s**	25GB	-
Qwen2.5-VL 7B							
↪ **rLiVS (Ours)**	**68.1**	4.0	56.1	3.6	2.7s	19GB	-

Offline Long VQA

Method	VS-Ego		VS-Movie		MovieChat		CG-Bench
	Acc.	Sco.	Acc.	Sco.	Acc.	Sco.	Acc.
Video-ChatGPT [22]	51.7	3.7	54.4	3.4	47.6	2.5	-
MovieChat [28]	52.2	3.4	39.1	2.3	62.3	3.2	-
Chat-UniVi [14]	50.9	3.8	54.0	3.4	-	-	25.9
LLaMA-VID [21]	54.8	3.9	51.4	3.4	53.2	3.8	-
Goldfish [1]	-	-	-	-	67.6	4.2	-
Flash-VStream-7B [38]	59.0	3.9	56.1	3.4	-	-	-
rLiVS (Ours)	**61.0**	3.9	**59.3**	3.6	**78.0**	4.0	**33.1**

Ablations

(a) Importance of recurrency

Method	RVS-Ego		RVS-Movie		MovieChat	
	Acc.	Sco.	Acc.	Sco.	Acc.	Sco.
rLiVS	**65.3**	**4.0**	**57.7**	**3.6**	**78.0**	**4.0**
w/o recurrency	62.5	3.9	53.7	3.5	74.1	3.9

(b) Answering Modality

Retrieved Modality	RVS-Ego		RVS-Movie	
	Acc.	Sco.	Acc.	Sco.
Selected Visual Tokens	58.2	3.9	48.4	3.5
Captions (ours)	**65.1**	4.0	**57.7**	3.6
Combination	63.0	4.0	54.3	3.5

Task Adaptability

Case Study: Instruction Sensitivity We selected two representative videos and evaluated the overlap of selected visual tokens using a generic captioning prompt versus task-specific instructions:

- **Video 1**: A girl with sunglasses is the main foreground object.
  - *Instruction*: "Track the locations of the girl wearing the sunglasses in the video."
  - ***Token Overlap with Generic Prompt*: 44%**
- **Video 2**: Two people playing cards, with interest focused on the background.
  - *Instruction*: "Locate and describe the objects appearing in the background of the video."
  - ***Token Overlap with Generic Prompt*: 8%** **Generic Prompt: "Describe what's happening in the video."

Summary

***rLiVS* focuses on**

- Continuity through recurrency
- Token selection through Model's attention
- Caption-based Answering

and results in…

SOTA on very long streaming VQA while being faster and more light–weight.

Limitations & Future Work

- FIFO memory may suffer from information drift
- Complement *rLiVS* with long-term memory that stores tokens evicted from the FIFO recent memory