StreamingVLM: Real-Time Understanding for Infinite Video Streams

Ruyi Xu*, Guangxuan Xiao*, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, Song Han

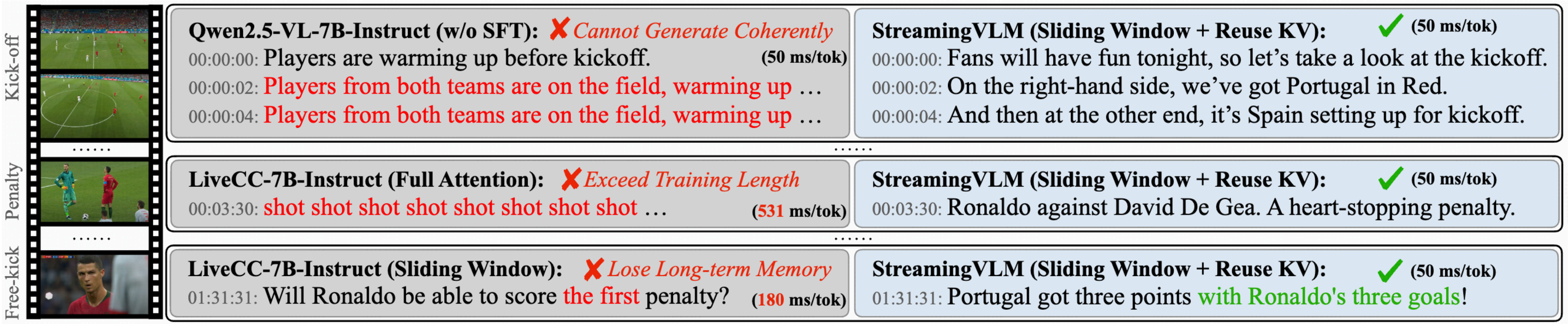
Massachusetts Institute of Technology
NVIDIA

*: Equal Contribution

Motivation

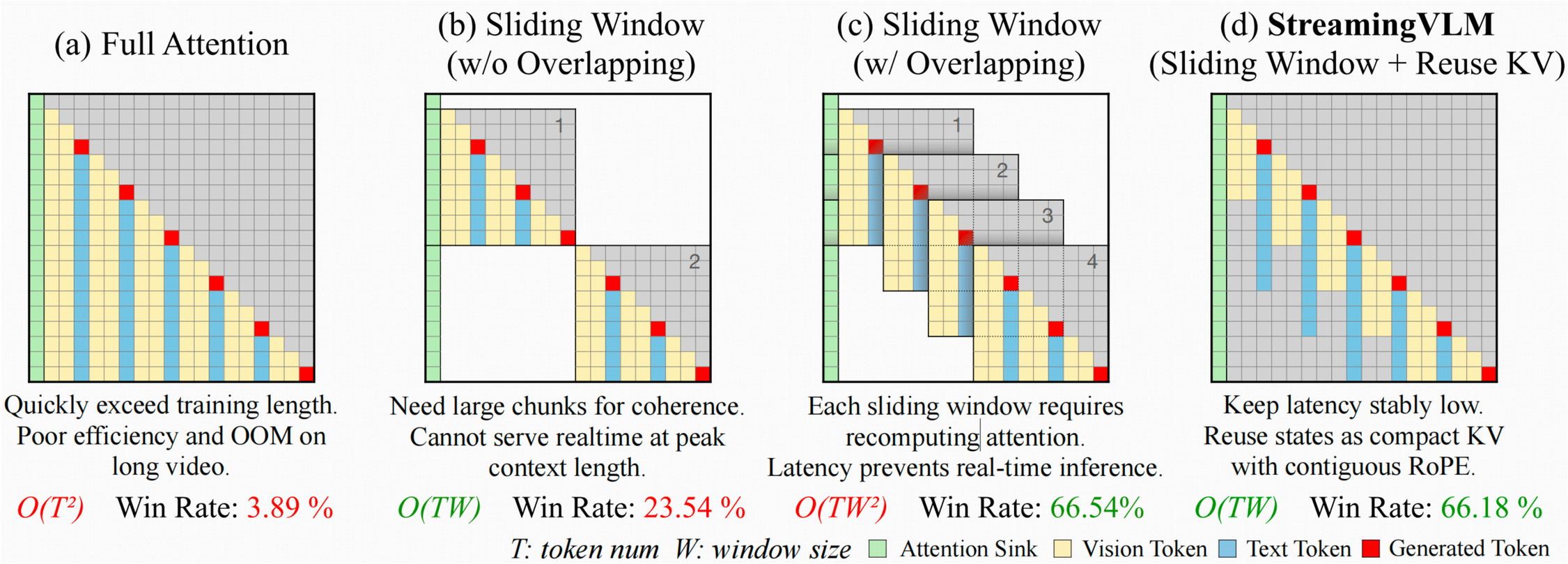
Understanding near-infinite video, responding in real time stably is challenging

- Processing long videos with full attention leads to quadratic computation and poor performance.
- Simple sliding window methods suffer from breaking coherence or redundant computational costs.
- Alignment between inference and training remains under-explored.



Key Challenges

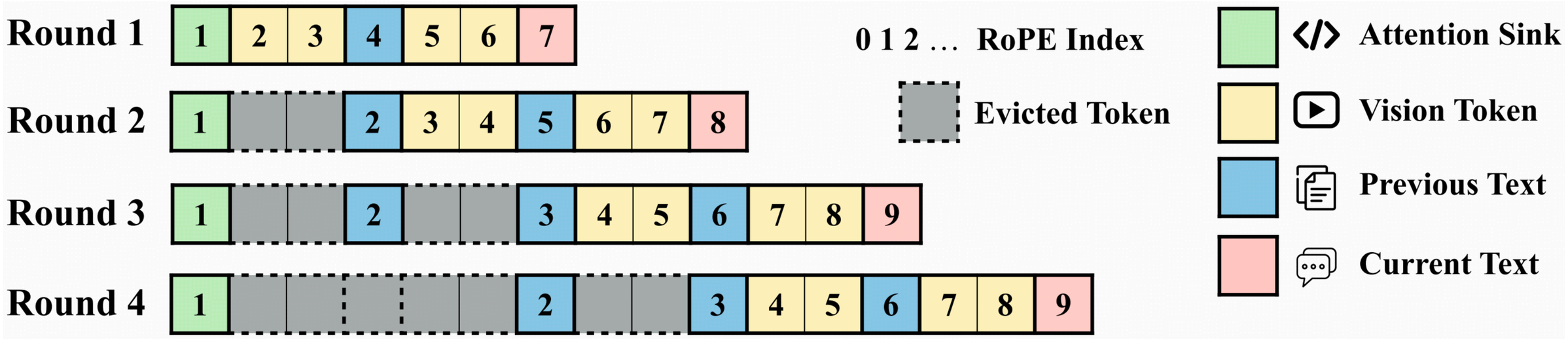
- **Full Attention:** $O(T^2)$ cost; unbounded memory; degrades beyond training length.
- **Sliding Window (w/o Overlapping):** short chunks break coherence; long chunks raise latency.
- **Sliding Window (w/ Overlapping):** recomputation per window yields high latency.



StreamingVLM: Real-Time Understanding for Infinite Video Streams

Inference Scheme

- **Streaming-aware KV Cache:**
  - Maintain a compact and stable KV cache by reusing previous states.
  - Older vision tokens are evicted first; early text is evicted only when the budget is exceeded.
- **Contiguous RoPE:**
  - Shift the RoPE indices so that their positions remain numerically contiguous.
  - The effective RoPE indices remain within a bounded range.



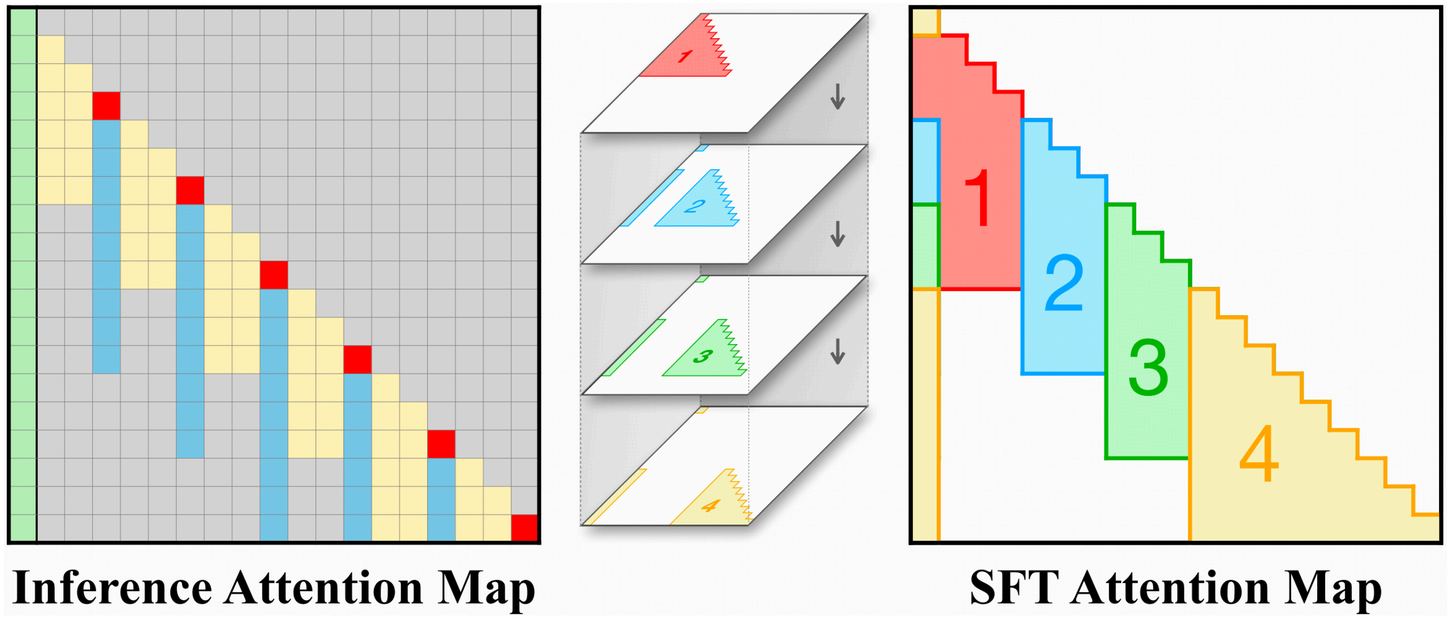
Training Strategy

Overlapped-chunk, Full-attention Strategy

- Overlapped full-attention supervision approximates the effective attention pattern at inference.

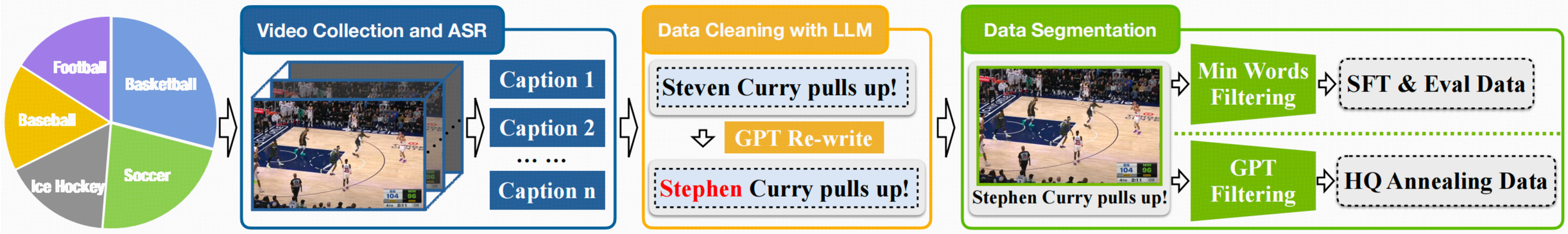
- **Overlapped-chunk:** Split a long video stream into consecutive overlapped chunks, rather than replicating the exact sliding-window schedule used for inference.
- **Full Attention:** Apply full attention within a chunk

.



Data Curation Pipeline

- **Video Collection and ASR:**
  - Collect game videos from five sports, averaging over 2 hours.
- **Data Cleaning:**
  - Request LLMs to decide “Keep”, “Delete”, or “Edit” a Caption.
- **SFT and Evaluation Data Segmentation:**
  - Split Videos with Overlapped-chunk
  - Create a new benchmark, Inf-Streams-Eval
- **High-quality Annealing Data:**
  - LLMs decide whether proportion of real-time commentary exceeds 80%

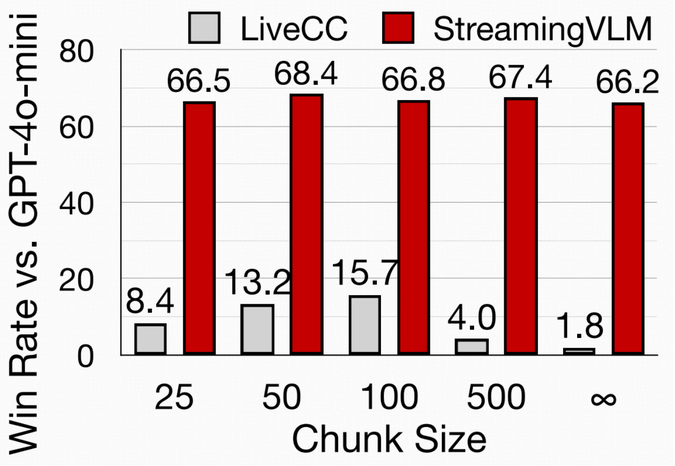


Results on Captioning Tasks

Captioning Benchmarks: Inf-Stream-Eval and Livecc-Sports-3K cc

- **StreamingVLM vs SFT methods:**

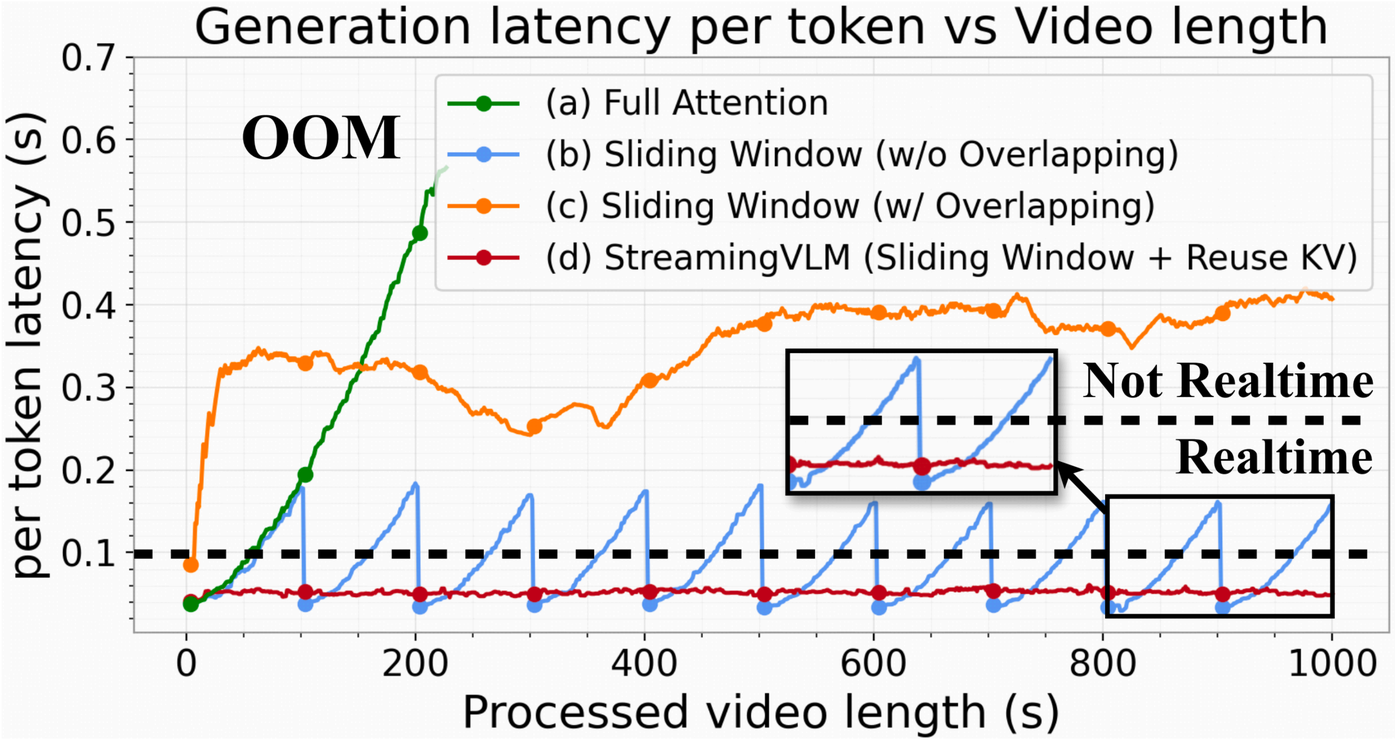
Win Rate A vs. B (Model A \ Model B)	Inf-Streams-Eval			Livecc-Sports-3K cc			
	GPT-4o†	Livecc†	Livecc∞	LLaVA	GPT-4o	Gemini	Livecc
Qwen-2.5-VL-7B-Instruct †	0.01	20.44	95.97	24.50	16.25	28.38	34.11
Livecc-7B-Instruct †	15.73	–	–	–	–	–	–
Livecc-7B-Instruct ∞	1.82	–	–	41.50	40.06	39.73	–
StreamingVLM ∞	**66.18**	**87.81**	**99.12**	**47.33**	**45.59**	**44.21**	**56.19**



- **StreamingVLM vs ReKV**
  - ReKV + non–fine-tuned model: lack the capability of real-time captioning,
  - ReKV + fine-tuning models: ReKV's eviction policy disrupts context, frequently resulting in no output.

Win Rate (Model A \ Model B)	Inf-Streams-Eval		
	GPT-4o†	Livecc†	Livecc∞
Qwen (+ ReKV) ∞	0.00	19.56	63.57
StreamingVLM (+ ReKV) ∞	0.00	0.00	0.00
StreamingVLM (+ Ours) ∞	**66.18**	**87.81**	**99.12**

Efficiency Results



- Full attention hits OOM;
- Sliding window w/o Overlapping spikes above real time;
- Sliding window w/ Overlapping remains inefficient;
- StreamingVLM latency stays low and stable.

Efficiency and Results on VQA Tasks

- **VQA:** Evaluate StreamingVLM and Qwen-2.5-VL-7B-Instruct, on 4 VQA tasks.
- Without any VQA SFT, StreamingVLM outperforms the base on all four tasks.

	MVBench	Video MME (w/o sub.)	LongVideoBench	OVOBench (Realtime)
Qwen-2.5-VL-7B-Instruct	67.34	65.10	54.70	56.00
StreamingVLM	**69.16**	**65.10**	**59.00**	**61.96**

Ablation Study

- **Contiguous RoPE**

Win Rate A vs. B (Model A \ Model B)	Inf-Streams-Eval		
	GPT-4o†	Livecc†	Livecc∞
Native †	63.23	74.00	98.07
Native ∞	25.09	59.42	60.32
Contiguous ∞	**66.18**	**87.81**	**99.12**

- **Sliding Window and Sink**

Infer args		SFT args		Inf-Streams-Eval (Basketball)		
T_{sink}	T_{window}	T_{sink}	T_{window}	GPT-4o†	Livecc†	Livecc∞
512	0	512	512	69.68	89.42	99.19
0	512	512	512	66.76	86.03	98.69
256	256	512	512	70.17	91.79	99.62
1024	1024	512	512	71.43	91.69	**99.84**
∞	∞	∞	∞	60.41	72.08	98.55
512	512	512	512	**73.64**	**92.33**	99.38

V_{window}	Inf-Streams-Eval		
Win Rate vs.	GPT-4o†	Livecc†	Livecc∞
0 s	52.90	77.49	97.56
1 s	63.46	83.24	98.18
4 s	66.08	83.86	98.73
8 s	65.66	85.09	99.14
32 s	65.49	85.58	99.06
16 s	**66.18**	**87.81**	**99.38**

- **Training Strategy and Dataset**

Win Rate A vs. B (Model A \ Model B)	Inf-Streams-Eval			Livecc-Sports-3K cc				MVBench	Video MME	LongVideoBench	OVOBench
	GPT-4o†	Livecc†	Livecc∞	LLaVA	GPT-4o	Gemini	Livecc Score		w/o sub.		Realtime
Qwen-2.5-VL-7B-Instruct †	0.01	20.44	95.97	24.50	16.25	28.38	34.11	67.34	**65.10**	54.70	56.00
+ Live-WhisperX-526K ∞	32.17	56.52	99.05	42.77	41.86	39.37	47.80	63.71	62.10	54.30	57.69
+ Inf-Streams-Train ∞	63.46	83.82	98.95	46.45	45.48	44.27	53.07	68.66	64.90	**59.00**	60.55
+ High-Quality Annealing Data ∞	**66.18**	**87.81**	**99.12**	**47.33**	**45.59**	**44.39**	**56.19**	**69.16**	**65.10**	**59.00**	**61.96**

Conclusion

- We present **StreamingVLM**, a model designed for **real-time, stable understanding of infinite visual input.**
- We present **a unified training–inference framework** that brings real-time streaming perception to existing VLMs.