ROMA: Real-time Omni-Multimodal Assistant with Interactive Streaming Understanding

Anonymous ACL submission

Abstract

Recent Omni-multimodal Large Language Models show promise in unified audio, vision, and text modeling. However, streaming audio-video understanding remains challenging, as existing approaches suffer from disjointed capabilities: they typically exhibit incomplete modality support or lack autonomous proactive monitoring. To address this, we present ROMA, **a real-time omni-multimodal assistant for unified reactive and proactive interaction.** ROMA processes continuous inputs as synchronized *multimodal units*, aligning dense audio with discrete video frames to handle granularity mismatches. For online decision-making, we introduce a lightweight *speak head* that decouples response initiation from generation to ensure precise triggering without task conflict. We train ROMA with a curated streaming dataset and a two-stage curriculum that progressively optimizes for streaming format adaptation and proactive responsiveness. To standardize the fragmented evaluation landscape, we reorganize diverse benchmarks into a unified suite covering both proactive (alert, narration) and reactive (QA) settings. Extensive experiments across 12 benchmarks demonstrate ROMA achieves state-of-the-art performance on proactive tasks while competitive in reactive settings, validating its robustness in unified real-time omni-multimodal understanding. Code and benchmark are available here[1].

1 Introduction

Recent advances in omni-multimodal large language models (OLLMs), such as GPT-4o (Hurst et al., 2024), have enabled unified modeling of speech, vision, and text. This progress facilitates real-world streaming audio-video understanding, defined as combining **reactive** and **proactive** capabilities (Figure 1). In the reactive setting, the model answers after the query, whereas in the proactive setting, it follows an instruction to continuously monitor the input stream and respond only when conditions are met. Unifying these capabilities is vital for real-world utility, yet the divergent interaction paradigms make it challenging (Horvitz, 1999; Xi et al., 2025; Driess et al., 2023).

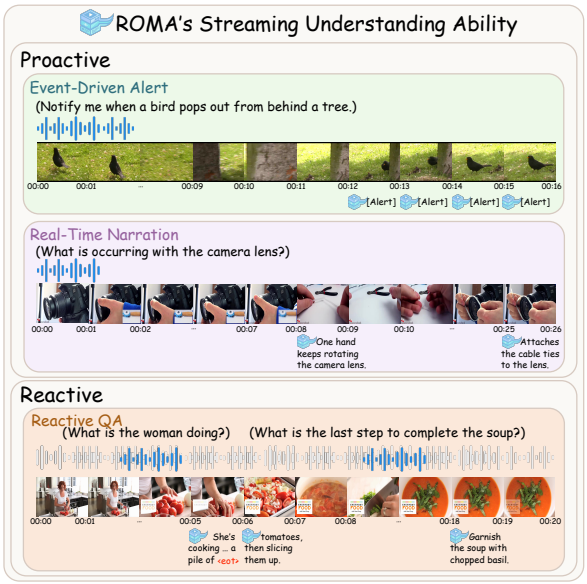


Figure 1: **ROMA's streaming understanding capabilities.** It supports proactive tasks, including event alerts and narration, alongside reactive question answering.

Despite the critical need for such unification, existing studies typically lack unified modality support and streaming capabilities. Specifically, speech-centric streaming models (Défossez et al., 2024; Zhang et al., 2025) focus on audio generation but lack visual perception. Conversely, while some approaches address streaming video understanding (Chen et al., 2024; Zhang et al., 2024b), they typically neglect synchronized audio and are confined to specific tasks (e.g., alert or narration). Consequently, unified streaming audio-video understanding remains largely under-explored.

To realize such unification faces two challenges.

[1] https://anonymous.4open.science/r/ROMA-6746

First, audio and video exhibit mismatched temporal granularities. While naturally synchronized, audio signals are dense and continuous, whereas video comprises sparse, discrete frames. Under such heterogeneity, maintaining robust cross-modal alignment and fusion demands precise synchronization. Second, effective streaming interaction requires real-time proactive decision-making. Upon integrating these asynchronous signals, the model must continuously synthesize context to determine both response timing and content, conditioned strictly on the stream prefix.

To address these challenges, we propose ROMA, a **R**eal-time **O**mni-**M**ultimodal **A**ssistant with interactive streaming understanding. To tackle the granularity mismatch, ROMA segments continuous audio into one-second intervals synchronized with video frames, forming temporally aligned units that are processed sequentially as the stream unfolds. We further adapt chunked Time-aligned Multimodal RoPE (TMRoPE) (Xu et al., 2025a) to enforce a shared temporal timeline. For proactive decision-making, ROMA introduces a lightweight speak head parallel to the standard language modeling (LM) head to explicitly predict response timing, decoupling timing from content generation to prevent task interference. Finally, we support this system with a custom streaming dataset and a two-stage training curriculum, progressively optimizing the model for cross-modal streaming format adaptation and proactive responsiveness.

For a comprehensive evaluation, streaming audio-video understanding demands assessing both reactive and proactive capabilities. However, as compared in Table 1, existing benchmarks suffer from inconsistent taxonomies and fragmented protocols, often failing to cover both interaction modes. To enable unified comparison, we reorganize the evaluation landscape into two standardized settings: a *proactive* mode that tests the ability to autonomously trigger responses at precise moments, and a *reactive* mode that emphasizes understanding temporal evolution in standard QA. Empirically, ROMA consistently outperforms existing streaming VideoLLMs across both modes. Furthermore, evaluations on open-ended audio-query QA against open-source OLLMs confirm its superior capability in unified audio-video understanding.

In summary, our contributions are as follows:

- **Unified streaming framework:** We formally define the task of streaming audio-video understanding and propose ROMA, an omni-multimodal assistant unifying reactive and proactive capabilities, supported by a curated dataset and a two-stage curriculum.
- **Standardized evaluation benchmark:** We establish a comprehensive streaming benchmark by reorganizing fragmented tasks into unified *reactive* and *proactive* settings to facilitate rigorous and consistent comparison.
- **Superior performance and analysis:** ROMA achieves state-of-the-art results across proactive benchmarks while competitive on reactive and open-ended QA. Extensive analysis verifies the efficacy of our timing mechanisms and training strategies.

Benchmark	Alert	Narration	Reactive QA
StreamingBench (Lin et al., 2024)	✓	✗	✓
StreamBench (Wu et al., 2024a)	✗	✗	✓
OVO-Bench (Niu et al., 2025)	✗	✓	✓
SVBench (Yang et al., 2025b)	✗	✗	✓
OmniMMI (Wang et al., 2025b)	✓	✗	✓
OVBench (Huang et al., 2025)	✗	✗	✓
Ours	✓	✓	✓

Table 1: Coverage of key streaming ability across representative streaming video benchmarks.

2 Related Works

Reactive Models Most existing streaming systems are studied in the reactive setting, answering only after the query arrives. Within this regime, memory-based methods maintain long-range context for coherent understanding over evolving streams (Qian et al., 2024; Zhang et al., 2024a; Wang et al., 2024b; Zhang et al., 2024b; Xiong et al., 2025; Wang et al., 2025a; Zhao et al., 2025), and KV-cache based methods optimize efficiency via scheduling or compression (Di et al., 2025; Ning et al., 2025; Yang et al., 2025a; Xu et al., 2025c; Chen et al., 2025b). Recent omni-multimodal models also adhere to this reactive protocol: MiniCPM-o 2.6 (Yao et al., 2024), Qwen2.5-Omni (Xu et al., 2025a), and Qwen3-Omni (Xu et al., 2025b) support low-latency interaction, and Stream-Omni (Zhang et al., 2025) enables visually-conditioned speech generation, yet none explicitly model proactive monitoring and triggering.

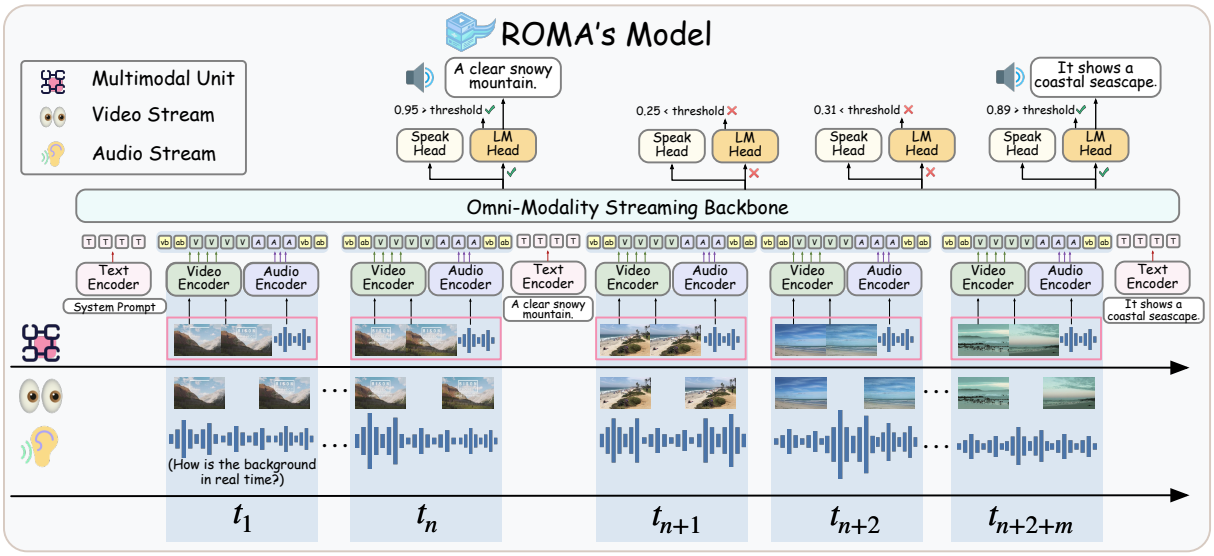


Figure 2: **Model Architecture.** Streaming inputs are processed as aligned multimodal units. The speak head determines response timing, activating the LM head (illustrated via narration) upon crossing a probability threshold.

Proactive Models In contrast, proactive streaming prioritizes continuous monitoring and time-sensitive triggering (e.g., alerts and real-time narration). Proactive VideoLLMs leverage online formats or explicit decision modeling to determine intervention timing (Chen et al., 2024; Yang et al., 2025d; Li et al., 2025; Yang et al., 2025c; Qian et al., 2025), with some explicitly targeting live narration (Chen et al., 2025a). However, these approaches remain predominantly video-centric, neglecting streaming audio.

Streaming Video Understanding Benchmarks Recent benchmarks prioritize time-sensitive, interactive evaluation (Table 1). StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025) assess temporal perception, while StreamBench (Wu et al., 2024a) and SVBench (Yang et al., 2025b) focus on long-horizon memory. Moreover, OmniMMI (Wang et al., 2025b) and OVBench (Huang et al., 2025) incorporate proactive capabilities, including real-time narration and alerts.

Tables 1 and 10 summarizes prior works.

3 Method

To unify reactive answering and proactive timing over continuous inputs, ROMA integrates architectural designs with a tailored training strategy. Section 3.1 presents the model architecture, utilizing chunked TMRoPE and a speak head for alignment and timing control. Section 3.2 details the training and inference pipeline, encompassing dataset construction and a two-stage fine-tuning recipe.

3.1 Model Architecture

As illustrated in Figure 2, ROMA processes streaming omni-modal inputs via a unified LLM backbone. We introduce a speak head parallel to the LM head to decouple interaction timing from content generation. This architecture addresses temporal alignment and proactive decision-making through the following mechanisms.

Multimodal units for temporally aligned streaming inputs. To support unified streaming understanding across modalities, we organize audio and video into fixed-interval multimodal units. Following the input format and tokenization of Qwen2.5-Omni, we treat all audio and video signals within each one-second interval as a unit. We align audio with video frames sampled from the same interval, extract their features, and wrap them with special tokens. This retains Qwen2.5-Omni's native format for compatibility while grounding audio in the preceding visual context:

<|vision_bos|><|audio_bos|>[*video tokens*][*audio tokens*]<|audio_eos|><|vision_eos|>

These multimodal units are fed into the LLM backbone sequentially as the stream unfolds. This process ensures that the model continuously accumulates aligned cross-modal context from the stream prefix, establishing a temporal basis for subsequent causal decision-making.

Chunk-Level Temporal Position Encoding We adapt Qwen2.5-Omni's Time-aligned Multimodal RoPE (TMRoPE) to chunked audio–video streams to support incremental encoding as units arrive.

Each one-second unit interleaves visual and auditory tokens, assigning time-aligned 3D position IDs (temporal, height, and width) to preserve their cross-modal correspondence. Consistent with the pre-trained vision encoder, multi-frame visual inputs are temporally aggregated into a fused representation during encoding. All video tokens within a unit therefore share a constant temporal ID. In contrast, audio tokens retain fine-grained temporal IDs at a 40ms resolution to preserve auditory temporal fidelity. To ensure boundary alignment, <|vision_bos|> and <|audio_bos|> share the same base position ID, subsequent units extend the global timeline by continuing from the maximum position ID of the previous unit.

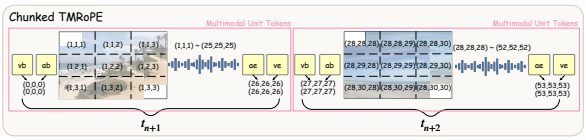


Figure 3: **Chunked TMRoPE.** Seamlessly extends the global timeline to streaming inputs by assigning cumulative positional IDs across discrete units.

Speak Head To enable autonomous intervention timing, we design a lightweight speak head. As illustrated in Figure 2, this module is implemented as a two-layer MLP, parallel to the LM head, on top of the streaming backbone. Upon processing each multimodal unit (one second of context), the speak head evaluates the current stream prefix and outputs a probability for a binary decision indicating whether a response is required. A response is triggered if this probability exceeds a threshold; otherwise, the model remains silent and continues consuming the stream. This design decouples the timing decision from text generation, mitigating interference from generative biases. Leveraging findings that upper layers encode high-level features (Tenney et al., 2019; Belrose et al., 2023), we compute the speak head input as a learnable weighted combination of hidden states from the last K layers, with $K{=}4$ in our experiments.

3.2 Training and Inference Pipeline

3.2.1 Dataset Construction

To enable end-to-end proactive and reactive supervision, we construct a comprehensive streaming dataset structured into two categories and three sub-tasks (Figure 4). Detailed processing pipelines are provided in Appendix A.4.

Online Proactive (27K) To equip the model with the ability to continuously monitor streams and trigger alerts, we curate data from DiDeMo (Anne Hendricks et al., 2017), OOPS (Epstein et al., 2020), and Charades-STA (Zhou et al., 2018). We reformulate these samples into alert-style tasks (e.g., "Alert me when [event] happens") to train the model in event-driven temporal grounding.

Online Narration (109K) To foster continuous event tracking and incremental summarization, we construct narration samples from MMDuetIT (Wang et al., 2024a), COIN (Tang et al., 2019), YouCook2 (Zhou et al., 2018), and ActivityNet (Caba Heilbron et al., 2015). Unlike prior works that use dense supervision, we specifically train the model to generate captions only at segment transitions, enabling it to provide concise, real-time updates as the visual context evolves.

Reactive QA (540K) To stabilize general audio–video understanding, we aggregate large-scale reactive QA data from InternVid (Wang et al., 2023), CogStream (Zhao et al., 2025), and others (Chen et al., 2023; Yang et al., 2022; Yao et al., 2025; Fu et al., 2025). These samples cover past events, temporal ordering, and future reasoning.

To ensure unified processing, we synthesize text queries into speech, training the model to handle audio instructions under streaming inputs.

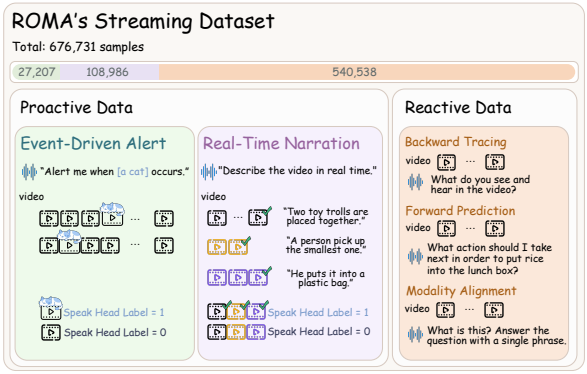


Figure 4: **Overview of ROMA's Streaming Dataset.** Statistics, task taxonomy, and sample formats.

3.2.2 A Two-Stage Fine-Tuning Recipe

Training an end-to-end streaming omni-multimodal model from scratch is computationally prohibitive. We fundamentally view streaming capability as a transfer problem: adapting a strong foundation model optimized for processing complete videos to handle incremental streams. We thus propose a simple yet effective two-stage recipe. Stage 1 adapts

the model to the streaming multimodal input format, while Stage 2 learns precise response timing and proactive policies. In both stages, we freeze all encoders and fine-tune the remaining parameters θ.

Stage 1: Streaming Template Alignment This stage mitigates the distribution shift between offline training and streaming inference. We utilize reactive QA datasets to adapt the model to the multimodal unit streaming format. Samples are restructured into sequential units X to simulate streaming, with the audio query and text response Y appended.

We optimize the standard autoregressive language modeling objective over the response tokens. Let $\mathcal{D}_{\text{QA}}$ denote the reactive QA dataset. For a sample $(X, Y) \sim \mathcal{D}_{\text{QA}}$, where $Y = \{y_1, \ldots, y_L\}$ represents the answer sequence, the loss is:

$$\mathcal{L}_{\text{LM}} = -\mathbb{E}_{(X,Y)\sim\mathcal{D}_{\text{QA}}} \left[\sum_{i=1}^{L} \log P(y_i \mid y_{<i}, X; \theta) \right].$$

This stage ensures the model retains its audio-video understanding while adapting to streaming inputs.

Stage 2: Time-Aware Decision Making With the backbone adapted to streaming inputs, this stage activates the speak head to learn *when* to respond. We formulate response timing as a binary classification task at each multimodal unit step. The positive labels are task-dependent: for proactive alerts, valid triggers lie within the event window; for narration, they align with segment boundaries. To mitigate trigger sparsity, we balance the loss using $w_{\text{pos}} = N_{\text{neg}}/N_{\text{pos}}$ derived from dataset statistics.

Let p_t be the speak head's predicted probability at time step t, and $z_t \in \{0, 1\}$ be the ground truth label. The timing loss is formulated as a weighted Binary Cross-Entropy (BCE):

$$\mathcal{L}_{\text{time}} = -\mathbb{E}_{X\sim\mathcal{D}_{\text{stream}}} \left[\frac{1}{T} \sum_{t=1}^{T} \Big(w_{\text{pos}}\, z_t \log p_t + (1 - z_t) \log(1 - p_t) \Big) \right].$$

To prevent generation quality degradation while optimizing purely for timing, we mix a small portion of the Stage 1 reactive QA data ($\mathcal{D}_{\text{QA}}$) during training. The final objective is a joint optimization:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{time}} + \lambda \cdot \mathcal{L}_{\text{LM}},$$

where $\mathcal{L}_{\text{LM}}$ is calculated only on the mixed QA samples to maintain linguistic competence, and λ balances the two objectives.

3.2.3 Inference Procedure

During inference, we strictly follow the training configuration. Video frames are uniformly sampled at 2 fps, and each frame is resized so that the number of pixels does not exceed 65,536. We maintain a persistent KV cache across the stream, so each step only encodes the current multimodal unit. Under this setup, encoding one unit takes 0.3697 seconds on average.

4 Unified Streaming Evaluation Framework

Effective streaming understanding demands models capable of answering queries and autonomously determining interaction timing. Addressing the fragmentation in existing benchmarks (Table 1), we establish a unified framework comprising two primary settings: **proactive interaction**, where the model autonomously monitors the stream to trigger responses, and **reactive interaction**, where it answers queries based on accumulated context.

4.1 Proactive Streaming Interaction

In the proactive setting, the model receives an instruction at the start and must process the stream to determine both the precise timing and content of the response. We categorize this into two sub-tasks: event-driven alert and real-time narration.

4.1.1 Event-Driven Alert

This task evaluates the model's temporal awareness, specifically its ability to detect transient events and trigger immediate notifications. We assess this capability under two settings.

Static Temporal Grounding. Following MMDuet on QVHighlights (Lei et al., 2021) and Charades-STA (Gao et al., 2017), ROMA incrementally predicts response probabilities for each multimodal unit. For QVHighlights, we rank timestamps by normalized probabilities, reporting mAP (ranking quality) and HIT@1 (top-1 accuracy). For localization on Charades-STA, we threshold probabilities to predict spans, reporting R@0.5 and R@0.7 (recall at 0.5 and 0.7 temporal overlap).

Dynamic Streaming Decision. This configuration enforces a strict streaming protocol where the model makes instantaneous decisions conditioned exclusively on the current multimodal unit. We conduct a comprehensive evaluation across OmniMMI (PA), StreamingBench (PO), and OVO-Bench (CRR, REC), spanning both single-event

alerts and multi-event recurrence. Specifically, for OVO-Bench, we reformulate the original QA-centric annotations into streaming alert targets to evaluate instantaneous responsiveness. To mitigate transient probability fluctuations, we employ a sliding window mechanism. Success is determined by the temporal inclusion of the autonomously triggered response within the ground-truth interval.

See Appendix A.3 for detailed settings.

4.1.2 Real-Time Narration

We define streaming narration as the incremental summarization of evolving events devoid of future context. To evaluate this capability, we employ two settings: a continuous YouCook2 adaptation, constructed by concatenating annotated clips to enforce generation at segment transitions, and the OVO-Bench (SSR) task, where responses are triggered via prediction thresholds and appended to the streaming context. Performance is assessed using the F1 score for temporal localization, BERTScore for the semantic quality of aligned responses, and a GPT-4o-based evaluation of coherence, alignment, and conciseness (detailed in Figure 12).

4.2 Reactive QA

In the reactive setting, the model must interpret temporal evolution to answer questions constrained to the causal video history. We utilize OVO-Bench and StreamingBench for standardized evaluation, employing text-based queries to ensure fairness against VideoLLMs baselines and reporting accuracy. To further approximate real-world interaction, we extend the assessment to Video-MME and EgoSchema using synthesized speech inputs. This setting evaluates comprehensive audio-video understanding, with open-ended responses scored by GPT-4o (detailed in Figure 11).

5 Experiment

5.1 Implementation Details

To address trigger sparsity, we set the positive weight $w_{\text{pos}} = 3$ in the weighted BCE loss. For inference, we adopt a pipelined real-time approximation: the model processes unit t while simultaneously acquiring unit $t+1$. To ensure synchronization, we cap generation at 25 tokens (approx. 1s) per segment, allowing longer responses to continue across subsequent units. Please refer to Appendix A.5 for detailed training configurations and complete decoding protocols.

5.2 Experimental Results

Baseline Methods In the proactive setting, we limit comparison to streaming-capable models: VideoLLM-Online (the basis for many efficiency-focused architectures), MMDuet, and Dispider. We reproduce results using accessible implementations, defaulting to reported figures otherwise. For Reactive QA, we benchmark against representative streaming VideoLLMs. To assess full-modality understanding, we extend evaluation to open-source omni-modal models, including Qwen2.5-Omni, MiniCPM-o, and VITA-1.5.

Method	QVHighlight	Charades-STA
	mAP / HIT@1	R@0.5 / 0.7
TimeChat	14.5 / 23.9	32.2 / 13.4
VTimeLLM	–	31.2 / 11.4
HawkEye	–	31.4 / 14.5
VTG-LLM	16.5 / 33.5	33.8 / 15.7
MMDuet	31.3 / 49.6	42.4 / 18.0
Ours	**53.7 / 53.0**	**44.3 / 19.9**
- Ablation Study		
Mixed Training	50.3 / 44.7	28.2 / 10.1
$K = 1$	46.4 / 47.4	32.4 / 13.1
- Sensitivity Analysis		
$w_{pos} = 2$	47.5 / 52.5	42.2 / 18.4
$w_{pos} = 4$	47.3 / 49.1	38.0 / 16.4

Table 2: Comparison with existing methods on QVHighlights and Charades-STA benchmarks.

Method	PA	PO	CRR	REC
VideoLLM-online	0.50	4.13	27.08	14.29
MMDuet	22.00	29.44	16.67	12.77
Dispider	–	25.34	**48.75**	18.05
M4-a	25.50	–	–	–
Ours	**37.50**	**53.60**	35.42	**33.81**
- Ablation Study				
Mixed Training	34.50	50.80	25.00	13.13
w/o Speak Head	12.50	12.00	0.00	6.46
$K = 1$	26.00	56.40	31.25	24.32
- Sensitivity Analysis				
$w_{pos} = 2$	31.00	52.76	39.58	31.54
$w_{pos} = 4$	31.00	52.15	37.50	26.74

Table 3: Comparison across single-alert (PA, PO, CRR) and recurring alert (REC) benchmarks.

Event-Driven Alert In static temporal grounding (Table 2), ROMA advances temporal localization on QVHighlights (53.7 mAP) and Charades-STA

(44.3/19.9 R@0.5/0.7), confirming that incremental speak probabilities provide enhanced temporal saliency for precise ranking and prediction. In the dynamic setting (Table 3), ROMA demonstrates strong efficacy on single-alert tasks: it excels on PA and PO while remaining competitive on CRR, validating its precise proactive triggering and robust evidence accumulation. Furthermore, ROMA dominates on the REC benchmark, validating its recurrence modeling for tracking repeated instances.

Real-Time Narration As shown in Table 4, ROMA achieves the best temporal triggering accuracy, obtaining an F1 score of 35.21 on YouCook2 and 14.54 on OVO-Bench (SSR), which indicates more precise alignment between generated responses and the annotated narration windows. It also achieves the highest GPT-4o score on both benchmarks. This score averages three criteria (story coherence, alignment to ground truth, and conciseness), with the per-criterion breakdown in Table 9, suggesting more coherent and better-aligned narration when generation is triggered online and the outputs are carried forward as context.

Method	YouCook2			OVO-Bench (SSR)		
	F1	BERT	GPT	F1	BERT	GPT
TimeChat	21.70	–	–	–	–	–
VTG-LLM	17.50	–	–	–	–	–
VideoLLM-online	18.82	0.82	0.17	10.24	**0.84**	0.18
MMDuet	17.81	0.83	0.23	9.02	0.79	0.31
Ours	**35.21**	**0.83**	**0.39**	**14.54**	0.83	**0.42**
- Ablation Study						
Mixed Training	31.42	0.81	0.34	8.88	0.80	0.33
w/o Speak Head	9.25	0.79	0.24	3.39	0.77	0.26
$K = 1$	34.43	0.82	0.37	9.64	0.78	0.32
- Sensitivity Analysis						
$w_{pos} = 2$	27.82	0.83	0.45	10.38	0.57	0.38
$w_{pos} = 4$	35.55	0.81	0.47	13.48	0.75	0.34

Table 4: Streaming narration results on YouCook2 and OVO-Bench (SSR). We report F1 for temporal window alignment, and use BERTScore and averaged GPT-4o scores to assess narration quality.

Reactive QA On OVO-Bench (Table 6), ROMA leads in both "Real-time Visual Perception" and "Backward Tracing". Its superiority over streaming baselines highlights enhanced sensitivity to time-localized cues and robust utilization of historical evidence under truncated contexts. On StreamingBench (Table 7), ROMA maintains high accuracy and secures the top rank on "Omni-Source Understanding" benchmark. attributed to preserving aligned audio during training, which bolsters audio–visual integration. In full-modality evaluation (Table 5), ROMA attains the best performance on Video-MME (without subtitles) and remains competitive on EgoSchema. Notably, these results utilize spoken queries with joint audio–visual inputs to approximate conversational interaction, distinct from text-prompted prior work.

Overall, ROMA strengthens temporal awareness and streaming decision-making, optimizing timing and content via audio–video joint modeling.

Method	Video-MME	EgoSchema
Qwen2.5-Omni	20.50	**58.40**
VITA-1.5	28.56	45.40
MiniCPM-o	19.37	55.20
Ours	**33.30**	55.40
- Ablation Study		
Mixed Training	33.00	50.20
w/o speak head	9.11	12.80
$K = 1$	34.56	54.00
- Sensitivity Analysis		
$w_{pos} = 2$	33.20	52.60
$w_{pos} = 4$	33.10	54.80

Table 5: Full-modality QA results on Video-MME (no subtitles) and EgoSchema, evaluated with spoken questions to approximate real conversational interaction.

5.3 Ablation Study

Single-Stage vs. Two-Stage Training We validate the two-stage curriculum by mixing all data and training directly with the stage-2 objective. This variant consistently degrades on tasks that require online timing and triggering, most notably on dynamic decision making (e.g., REC) and streaming narration (Table 3, Table 4). The results indicate that progressive training is important for learning well-calibrated temporal decision making under streaming input.

Speak Head for Response Gating We replace the speak head with a '<|silence|>' token following prior work, and cast triggering as next-token prediction with a reweighted loss. Lacking explicit probabilities, we omit QVHighlights and Charades-STA, instead evaluating triggering based on the first non-'<|silence|>' token.

Method	Real-time Visual Perception						Backward Tracing		
	OCR	ACR	ATR	STU	FPD	OJR	EPM	ASI	HLD
VideoLLM-online	8.05	23.85	12.07	14.04	45.54	21.20	22.22	18.80	12.18
MMDuet	13.42	11.93	14.66	11.80	14.85	10.33	10.44	8.78	0.54
Dispider	57.72	49.54	62.07	**44.94**	61.39	51.63	48.48	**55.41**	4.30
Flash-VStream-7B	24.16	29.36	28.45	33.71	25.74	28.80	39.06	37.16	5.91
Ours	**63.09**	**53.21**	**68.10**	39.33	**69.31**	**58.15**	**55.89**	47.30	**23.66**
- Ablation Study									
Mixed Training	63.09	55.05	63.79	37.64	61.39	55.43	55.22	45.95	27.96
w/o Speak Head	61.07	55.05	63.97	39.89	65.35	54.89	53.87	47.97	29.03
$K = 1$	61.47	55.05	68.10	39.89	65.35	60.33	56.57	46.62	20.97
- Sensitivity Analysis									
$w_{pos} = 2$	64.43	51.38	68.97	39.33	64.36	60.87	54.88	46.62	20.97
$w_{pos} = 4$	65.10	54.13	68.97	38.20	70.30	61.41	56.57	46.27	22.58

Table 6: Reactive QA results on OVO-Bench (excluding Forward Active Responding), evaluating time-sensitive understanding across Real-time Visual Perception and Backward Tracing.

Method	Real-Time Visual Understanding										Omni-Source Understanding				Contextual Understanding		
	OP	CR	CS	ATP	EU	TR	PR	SU	ACP	CT	ER	SCU	SD	MA	ACU	MCU	SQA
VideoLLM-Online	39.07	40.06	34.49	31.05	45.96	32.40	31.48	34.16	42.49	27.89	31.20	26.51	24.10	32.00	24.19	29.20	26.55
Flash-VStream	25.89	43.57	24.91	23.87	27.33	13.08	18.52	25.20	23.87	48.70	25.91	24.90	25.60	28.40	24.80	25.20	24.12
Dispider	74.92	75.53	74.10	73.08	74.44	59.52	76.14	**62.91**	62.16	45.80	35.46	25.26	38.57	43.34	**39.62**	27.65	33.61
Ours	**76.96**	**78.91**	**77.92**	**82.05**	**74.84**	**72.90**	**82.41**	61.79	**65.91**	**51.06**	**40.40**	**34.80**	**50.40**	**58.80**	37.60	**34.00**	**44.47**
- Ablation Study																	
Mixed Training	75.51	85.71	76.19	78.23	59.77	61.05	73.21	60.00	59.67	23.38	38.80	26.00	40.80	47.20	35.60	27.20	26.80
w/o Speak Head	76.13	70.49	74.14	82.40	72.86	70.80	84.78	63.20	64.91	51.69	39.75	30.36	45.87	47.20	35.27	24.79	24.80
$K = 1$	76.69	82.03	78.86	82.05	74.84	72.90	79.63	59.76	64.49	50.53	38.80	29.60	46.40	51.60	33.20	27.60	22.80
- Sensitivity Analysis																	
$w_{pos} = 2$	75.61	81.25	76.97	82.37	71.70	75.08	81.48	62.20	65.62	50.00	39.60	30.80	59.38	52.40	35.60	28.00	26.40
$w_{pos} = 4$	75.61	80.47	79.18	82.37	73.58	75.08	82.41	63.10	65.34	47.34	39.60	28.80	44.40	51.60	35.60	28.40	26.40

Table 7: Reactive QA results on StreamingBench (excluding PO), evaluating real-time understanding under streaming input across Real-Time Visual Understanding, Omni-Source Understanding, and Contextual Understanding.

Last-Layer vs. Last-4-Layer Aggregation We ablate four-layer aggregation by restricting the speak head to the final layer (K=1). This notably degrades temporal grounding and dynamic triggering (Tables 2, 3) while leaving timestamp-conditioned understanding largely unaffected (Tables 6, 7). This confirms multi-layer aggregation yields robust signals essential for streaming.

5.4 Sensitivity analysis

We sweep the positive weight w_{pos} in the weighted BCE loss of the speak head to mitigate the class imbalance from sparse speaking timestamps. We observe that w_{pos} is critical for proactive tasks (Tables 2–4), while reactive understanding and full-modality QA remain insensitive (Tables 5, 6). Overall, $w_{\text{pos}} = 3$ yields the most balanced performance. See Appendix A.2 for sensitivity analysis on inference-time triggering thresholds.

6 Conclusion

We introduce ROMA, a real-time omni-multimodal assistant that redefines streaming interaction as the unification of proactive and reactive paradigms. ROMA is the first framework to excel in both modes. To achieve this, we construct a streaming dataset and training recipe that enhance temporal modeling and decision-making. Furthermore, we standardize evaluation through a unified protocol tailored to this dual paradigm, where ROMA demonstrates superior performance. Finally, we provide a systematized overview of prior methods to facilitate future research.

Limitations

While optimized for streaming interaction, the model remains susceptible to distortions such as signal degradation and audio–video asynchrony. Additionally, while capable of continuous streaming, capturing extremely long-term dependencies spanning hours remains constrained by finite context windows and memory. Finally, optimizing the trade-off between inference efficiency and response quality under strict resource constraints remains a critical direction for future work.

Ethical Statement

This work utilizes publicly available datasets consistent with their original licenses. While ROMA enables proactive monitoring capabilities, we acknowledge the potential risk of misuse for unauthorized surveillance or privacy infringement. This model is intended for research purposes; due to the possibility of hallucinations or biases inherited from the base LLM, human oversight is strictly required for critical real-world applications.

References

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. 2017. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pages 5803–5812.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. 2024. Videollm-online: Online video large language model for streaming video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18407–18418.

Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. 2025a. Livecc: Learning video llm with streaming speech transcription at scale. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29083–29095.

Yi Chen, Yuying Ge, Yixiao Ge, Mingyu Ding, Bohao Li, Rui Wang, Ruifeng Xu, Ying Shan, and Xihui Liu. 2023. Egoplan-bench: Benchmarking multimodal large language models for human-level planning. *arXiv preprint arXiv:2312.06722*.

Yilong Chen, Xiang Bai, Zhibin Wang, Chengyu Bai, Yuhan Dai, Ming Lu, and Shanghang Zhang. 2025b. Streamkv: Streaming video question-answering with segment-based kv cache retrieval and compression. *arXiv preprint arXiv:2511.07278*.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Tao Zhong, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. 2025. Streaming video question-answering with in-context video kv-cache retrieval. *arXiv preprint arXiv:2503.00540*.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, and 1 others. 2023. Palm-e: An embodied multimodal language model.

Dave Epstein, Boyuan Chen, and Carl Vondrick. 2020. Oops! predicting unintentional action in video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 919–929.

Shenghao Fu, Qize Yang, Yuan-Ming Li, Yi-Xing Peng, Kun-Yu Lin, Xihan Wei, Jian-Fang Hu, Xiaohua Xie, and Wei-Shi Zheng. 2025. Vispeak: Visual instruction feedback in streaming videos. *arXiv preprint arXiv:2503.12769*.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275.

Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 159–166.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. 2025. Online video understanding: Ovbench and videochat-online. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3328–3338.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jie Lei, Tamara L Berg, and Mohit Bansal. 2021. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858.

Wei Li, Bing Hu, Rui Shao, Leyang Shen, and Liqiang Nie. 2025. Lion-fs: Fast & slow video-language thinker as online video assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3240–3251.

Junming Lin, Zheng Fang, Chi Chen, Zihao Wan, Fuwen Luo, Peng Li, Yang Liu, and Maosong Sun. 2024. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. *arXiv preprint arXiv:2411.03628*.

Zhenyu Ning, Guangda Liu, Qihao Jin, Wenchao Ding, Minyi Guo, and Jieru Zhao. 2025. Livevlm: Efficient online video understanding via streaming-oriented kv cache and retrieval. *arXiv preprint arXiv:2505.15269*.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, and 1 others. 2025. Ovobench: How far is your video-llms from real-world online video understanding? In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18902–18913.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. 2025. Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24045–24055.

Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. 2024. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360.

ByteDance Seed, Jiaze Chen, Tiantian Fan, Xin Liu, Lingjun Liu, Zhiqi Lin, Mingxuan Wang, Chengyi Wang, Xiangpeng Wei, Wenyuan Xu, and 1 others. 2025. Seed1. 5-thinking: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*.

Yemin Shi, Yu Shu, Siwei Dong, Guangyi Liu, Jaward Sesay, Jingwen Li, and Zhiting Hu. 2025. Voila: Voice-language foundation models for real-time autonomous interaction and voice role-play. *arXiv preprint arXiv:2505.02707*.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. 2019. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*.

Haibo Wang, Bo Feng, Zhengfeng Lai, Mingze Xu, Shiyu Li, Weifeng Ge, Afshin Dehghan, Meng Cao, and Ping Huang. 2025a. Streambridge: Turning your offline video large language model into a proactive streaming assistant. *arXiv preprint arXiv:2505.05467*.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, and 1 others. 2023. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*.

Yueqian Wang, Xiaojun Meng, Yuxuan Wang, Jianxin Liang, Jiansheng Wei, Huishuai Zhang, and Dongyan Zhao. 2024a. Videollm knows when to speak: Enhancing time-sensitive video comprehension with video-text duet interaction format. *arXiv preprint arXiv:2411.17991*.

Yuxuan Wang, Yueqian Wang, Bo Chen, Tong Wu, Dongyan Zhao, and Zilong Zheng. 2025b. Omnimmi: A comprehensive multi-modal interaction benchmark in streaming video contexts. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18925–18935.

Yuxuan Wang, Cihang Xie, Yang Liu, and Zilong Zheng. 2024b. Videollamb: Long-context video understanding with recurrent memory bridges. *arXiv preprint arXiv:2409.01071*.

Cheng-Kuang Wu, Zhi Rui Tam, Chieh-Yen Lin, Yun-Nung Vivian Chen, and Hung-yi Lee. 2024a. Streambench: Towards benchmarking continuous improvement of language agents. *Advances in Neural Information Processing Systems*, 37:107039–107063.

Shiwei Wu, Joya Chen, Kevin Qinghong Lin, Qimeng Wang, Yan Gao, Qianli Xu, Tong Xu, Yao Hu, Enhong Chen, and Mike Zheng Shou. 2024b. Videollm-mod: Efficient video-language streaming with mixture-of-depths vision computation. *Advances in Neural Information Processing Systems*, 37:109922–109947.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, and 1 others. 2025. The rise and potential of large language model based agents: A survey. *Science China Information Sciences*, 68(2):121101.

Haomiao Xiong, Zongxin Yang, Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Jiawen Zhu, and Huchuan Lu. 2025. Streaming video understanding and multi-round interaction with memory-enhanced knowledge. *arXiv preprint arXiv:2501.13468*.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, and 1 others. 2025a. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, and 19 others. 2025b. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*.

Ruyi Xu, Guangxuan Xiao, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, and Song Han. 2025c. Streamingvlm: Real-time understanding for infinite video streams. *arXiv preprint arXiv:2510.09608*.

Haolin Yang, Feilong Tang, Lingxiao Zhao, Xiang An, Ming Hu, Huifa Li, Xinlin Zhuang, Yifan Lu, Xiaofeng Zhang, Abdalla Swikir, and 1 others. 2025a. Streamagent: Towards anticipatory agents for streaming video understanding. *arXiv preprint arXiv:2508.01875*.

Pinci Yang, Xin Wang, Xuguang Duan, Hong Chen, Runze Hou, Cong Jin, and Wenwu Zhu. 2022. Avqa: A dataset for audio-visual question answering on videos. In *Proceedings of the 30th ACM international conference on multimedia*, pages 3480–3491.

Zhenyu Yang, Yuhang Hu, Zemin Du, Dizhan Xue, Shengsheng Qian, Jiahong Wu, Fan Yang, Weiming Dong, and Changsheng Xu. 2025b. Svbench: A benchmark with temporal multi-turn dialogues for streaming video understanding. *arXiv preprint arXiv:2502.10810*.

Zhenyu Yang, Kairui Zhang, Yuhang Hu, Bing Wang, Shengsheng Qian, Bin Wen, Fan Yang, Tingting Gao, Weiming Dong, and Changsheng Xu. 2025c. Livestar: Live streaming assistant for real-world online video understanding. *arXiv preprint arXiv:2511.05299*.

Zhiwei Yang, Chen Gao, Jing Liu, Peng Wu, Guansong Pang, and Mike Zheng Shou. 2025d. Assistpda: An online video surveillance assistant for video anomaly prediction, detection, and analysis. *arXiv preprint arXiv:2503.21904*.

Linli Yao, Yicheng Li, Yuancheng Wei, Lei Li, Shuhuai Ren, Yuanxin Liu, Kun Ouyang, Lean Wang, Shicheng Li, Sida Li, and 1 others. 2025. Timechat-online: 80% visual tokens are naturally redundant in streaming videos. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 10807–10816.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, and 1 others. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*.

Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, Jifeng Dai, and Xiaojie Jin. 2024a. Flash-vstream: Memory-based real-time understanding for long video streams. *arXiv preprint arXiv:2406.08085*.

Pan Zhang, Xiaoyi Dong, Yuhang Cao, Yuhang Zang, Rui Qian, Xilin Wei, Lin Chen, Yifei Li, Junbo Niu, Shuangrui Ding, and 1 others. 2024b. Internlm-xcomposer2. 5-omnilive: A comprehensive multimodal system for long-term streaming video and audio interactions. *arXiv preprint arXiv:2412.09596*.

Shaolei Zhang, Shoutao Guo, Qingkai Fang, Yan Zhou, and Yang Feng. 2025. Stream-omni: Simultaneous multimodal interactions with large language-vision-speech model. *arXiv preprint arXiv:2506.13642*.

Zicheng Zhao, Kangyu Wang, Shijie Li, Rui Qian, Weiyao Lin, and Huabin Liu. 2025. Cogstream: Context-guided streaming video question answering. *arXiv preprint arXiv:2506.10516*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Luowei Zhou, Chenliang Xu, and Jason Corso. 2018. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

A Appendix

A.1 Related Works

To address the fragmented landscape of streaming multimodal models, we unify representative methods in a comparative analysis along two axes: supported input modalities and interaction capabilities. We observe that many works described as "streaming" in fact adopt a question-injection protocol, where a query is issued at a predetermined timestamp and the model answers using only the preceding context. As a result, they primarily study long-horizon processing via KV-cache compression and external memory, rather than continuous online interaction with response-timing decisions. In contrast, the few systems that support online streaming interaction typically span all three interaction types. LiveCC is a notable exception: it focuses on fixed-rate real-time narration and therefore does not require deciding when to respond. Moreover, LION-FS, VideoLLM-MoD, and LiveStar mainly introduce efficiency improvements on top of the VideoLLM-online pipeline. Accordingly, we use VideoLLM-online as the representative baseline. Overall, Table 10 shows that our method is the first open-source model to enable full omni-modal streaming while natively supporting proactive response, real-time narration, and reactive QA within a unified framework.

We also summarize commonly used benchmarks for streaming evaluation in Table 1. Although these benchmarks are often described as "streaming", they target different capabilities, and their coverage is uneven, which motivates us to consolidate them into a unified evaluation protocol.

A.2 Sensitivity Analysis

Sensitivity analysis confirms robust performance (Figure 5). In static settings, mAP remains stable while HIT@1 shows only slight sensitivity to variations in the window size. Dynamic tasks exhibit a broad operating regime with smooth degradation, indicating no brittle reliance on specific parameters. Narration is likewise insensitive to speak head probability thresholds, justifying a fixed default without additional tuning (Table 8).

A.3 Evaluation Details

We specify evaluation protocols for our streaming interaction tasks. For PO, we preprocess each sample to replicate the original benchmark by cropping the video to the annotated ask time and injecting the question at that timestamp, ensuring strictly causal temporal ordering. While Streaming VLM baselines take text prompts, our model takes a speech rendering of the same text to benchmark native multimodal processing. For streaming baselines (e.g., VideoLLM-Online and MMDuet), we record the first-response timestamp and report accuracy as the fraction of samples whose first-response time is within ±2 seconds of the annotated ground-truth time. For REC, a model gets one point if its chosen response time falls within the annotated event interval. Each segment is evaluated once, and we report the micro success rate. For CRR, we award one point if the first response after the ask time occurs after the annotated clue time, validating that the model waits for necessary visual evidence before answering. Hyperparameters were determined via validation sets as follows: QVHighlight window size = 5; Charades-STA window size = 3 with threshold = 0.45; PA window size = 5 with threshold = 0.5; PO window size = 4 with threshold = 0.2; REC window size = 2 with threshold = 0.7; CRR window size = 2 with threshold = 0.7; YouCook2 threshold = 0.975; SSR threshold = 0.97.

Due to space constraints, in the main table we report only the average score on the narration task, computed as the mean of the three GPT-4o–based evaluation dimensions. We present the full breakdown in Table 9.

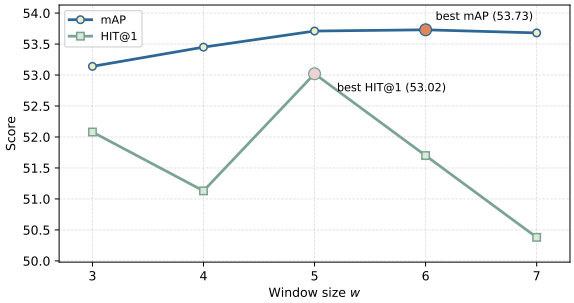


Figure 5: Sensitivity analysis on window size on QVHighlight.

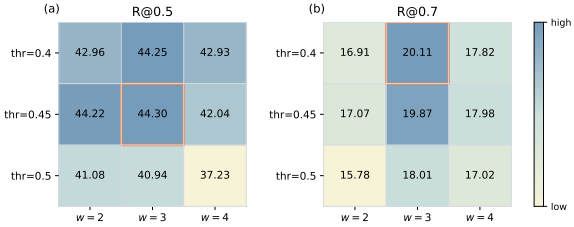


Figure 6: Sensitivity analysis on window size and threshold on Charades-STA.

Probability Threshold	YouCook2			OVO-Bench (SSR)		
	F1	BERTScore	GPT-Eval	F1	BERTScore	GPT-Eval
0.965	35.36	0.82	0.52 / 0.28 / 0.31	**15.53**	0.82	0.59 / 0.28 / 0.29
0.970	35.05	0.82	0.50 / 0.29 / 0.33	14.54	0.83	**0.59 / 0.33 / 0.34**
0.975	**35.21**	**0.83**	**0.53 / 0.29 / 0.36**	14.58	0.83	0.62 / 0.32 / 0.33
0.980	34.90	0.83	0.55 / 0.28 / 0.36	15.15	**0.84**	0.59 / 0.33 / 0.36
0.985	34.07	0.83	0.52 / 0.29 / 0.31	14.73	0.84	0.62 / 0.29 / 0.35

Table 8: Sensitivity analysis of the probability threshold on real-time narration, presenting performance metrics (F1, BERTScore, GPT-Eval) across different triggering thresholds on YouCook2 and OVO-Bench (SSR).

Method	YouCook2			SSR		
	F1	BERTScore	GPT-Eval	F1	BERTScore	GPT-Eval
TimeChat	21.70	–	–	–	–	–
VTG-LLM	17.50	–	–	–	–	–
VideoLLM-online	18.82	0.82	0.33 / 0.05 / 0.12	10.24	**0.84**	0.39 / 0.02 / 0.14
MMDuet	17.81	0.83	0.31 / 0.26 / 0.12	9.02	0.79	0.42 / 0.29 / 0.21
Ours	**35.21**	**0.83**	**0.53 / 0.29 / 0.36**	**14.54**	0.83	**0.59 / 0.33 / 0.34**
- Ablation Study						
Mixed Training	31.42	0.81	0.47 / 0.30 / 0.24	8.88	0.80	0.52 / 0.34 / 0.13
w/o Speak Head	9.25	0.79	0.32 / 0.32 / 0.09	3.39	0.77	0.41 / 0.30 / 0.08
$K = 1$	34.43	0.82	0.51 / 0.30 / 0.31	9.64	0.78	0.49 / 0.24 / 0.22
- Sensitivity Analysis						
$w_{pos} = 2$	27.82	0.83	0.62 / 0.27 / 0.46	10.38	0.57	0.63 / 0.21 / 0.29
$w_{pos} = 4$	35.55	0.81	0.64 / 0.27 / 0.49	13.48	0.75	0.54 / 0.20 / 0.28

Table 9: Streaming narration results on YouCook2 and OVO-Bench (SSR). We report F1 for temporal window alignment, and use BERTScore and GPT-4o scores to assess narration quality.

A.4 Data Construction Details

Proactive Data Processing Since the original temporal annotations in DiDeMo and Charades-STA are often coarse, simply using them for streaming supervision introduces noise. We therefore re-annotate event windows using Doubao-Seed-1.6-thinking (Seed et al., 2025) to obtain precise start and end timestamps. For timing supervision, we label every second within the refined ground-truth event window as a positive trigger, ensuring the model learns robust event sensitivity.

Narration Data Processing Raw videos often contain unlabeled gaps that induce hallucination during training. We mitigate this by excising these intervals and concatenating annotated segments into continuous, semantically dense streams with recalibrated timestamps. For timing supervision, we discard the broad-window labeling of prior work (e.g., MMDuetIT) in favor of strict transition-based triggering. This yields precise supervision for incremental narration via multi-turn SFT.

Audio Query Synthesis To simulate real-world interaction, we synthesize text queries via TTS and overlay them onto original audio tracks. We strictly align these spoken queries with streaming units to enforce audio-driven instruction following.

A.5 Implementation Details

Training Configuration. We sample videos at 2 FPS and resize frames to a maximum of 65,536 pixels. The model is trained using LLaMA-Factory (Zheng et al., 2024) with a sequence length of 32K on 32 H20 GPUs (using a global batch size of 512). Proactive samples are specifically formatted as multi-turn dialogues to handle multiple triggers within a single stream.

Streaming Decoding Logic. Following the pipelined setting, if a response exceeds the 25-token budget, we append an <|endoftext|> (<eot>) token to signal an unfinished utterance. Decoding resumes in the subsequent segment and terminates only when <|im_end|> is generated.

Method	Inputs	Alert	Narration	Reactive QA	Description
Moshi (Défossez et al., 2024)	A	✗	✗	✓	A full-duplex speech–text model enabling low-latency, real-time voice dialogue without vision.
Voila (Shi et al., 2025)	A	✗	✗	✓	Real-time voice-language model with expressive role-play and vocal styles.
VideoStreaming (Qian et al., 2024)	T+V	✗	✗	✓	Streaming framework with hierarchical memory for coherent long-video understanding.
Flash-VStream (Zhang et al., 2024a)	T+V	✗	✗	✓	Real-time system with lightweight memory for low-latency long video processing.
VideoLLaMB (Wang et al., 2024b)	T+V	✗	✗	✓	Long-context model with recurrent memory bridges to propagate information over time.
InternLM-XComposer2.5 (Zhang et al., 2024b)	T+V	✗	✗	✓	Multimodal streaming system with layered memory for long-term video–audio interaction.
StreamChat (Xiong et al., 2025)	T+V	✗	✗	✓	Streaming agent using hierarchical memory to sustain long-context, multi-round dialogue.
StreamBridge (Wang et al., 2025a)	T+V	✗	✗	✓	Plug-and-play buffer turning offline Video-LLMs into proactive streaming assistants.
CogStream (Zhao et al., 2025)	T+V	✗	✗	✓	QA framework keeping only memory-critical context for efficient reasoning.
ReKV (Di et al., 2025)	T+V	✗	✗	✓	Retrieves historical KV-cache as in-context memory instead of re-encoding past frames.
LiveVLM (Ning et al., 2025)	T+V	✗	✗	✓	Optimizes streaming KV-cache update/retrieval for efficient long-horizon processing.
StreamAgent (Yang et al., 2025a)	T+V	✗	✗	✓	Uses KV-based temporal memory to anticipate events and respond proactively.
StreamingVLM (Xu et al., 2025c)	T+V	✗	✗	✓	Manages rolling KV-cache to support infinite video streams under bounded computation.
StreamKV (Chen et al., 2025b)	T+V	✗	✗	✓	Segments and compresses KV-cache, keeping only salient past segments within budget.
LiveCC (Chen et al., 2025a)	T+V	✗	✓	✗	Trained with continuous speech transcription for real-time narration of long videos.
VideoLLM-online (Chen et al., 2024)	T+V	✓	✓	✓	Online model for temporally aligned dialogue via a LIVE training framework.
AssistPDA (Yang et al., 2025d)	T+V	✓	✓	✓	Surveillance assistant unifying anomaly prediction and interactive analysis.
LION-FS (Li et al., 2025)	T+V	✓	✓	✓	Uses fast–slow thinking with selective tokenization for efficient streaming.
VideoLLM-MoD (Wu et al., 2024b)	T+V	✓	✓	✓	Enables efficient streaming by letting each layer skip a subset of tokens directly.
LiveStar (Yang et al., 2025c)	T+V	✓	✓	✓	Uses perplexity-based timing and streaming-aware attention for proactive understanding.
MMDuet (Wang et al., 2024a)	T+V	✓	✓	✓	Adopts a video–text duet format to insert replies during continuous playback.
Dispider (Qian et al., 2025)	T+V	✓	✓	✓	Disentangles perception, decision, and reaction for asynchronous responses.
MiniCPM-o 2.6 (Yao et al., 2024)	T+V+A	✗	✗	✓	Omni model supporting low-latency streaming speech over text, audio, and video.
Qwen2.5-Omni (Xu et al., 2025a)	T+V+A	✗	✗	✓	Dense Thinker–Talker model with streaming encoders for real-time perception.
Qwen3-Omni (Xu et al., 2025b)	T+V+A	✗	✗	✓	Natively omni-modal MoE architecture for high-concurrency streaming generation.
Stream-Omni (Zhang et al., 2025)	T+V+A	✗	✗	✓	Efficient modality alignment enabling speech interaction grounded on visual inputs.
ViSpeak (Fu et al., 2025)	T+V+A	✓	✗	✓	Vision-centric framework producing instruction-like feedback from evolving streams.
ROMA (Ours)	T+V+A	✓	✓	✓	**End-to-end streaming OLLM processing time-aligned chunks for proactive interaction.**

Table 10: Comparison of different streaming multimodal methods. **Note:** T=Text, V=Visual, A=Audio. ✓ supports the ability, ✗ does not.

A.6 Case Study

Event-Triggered Alert We present two event-triggered alert cases: one where the target event occurs only once (Figure 7), and another where it recurs multiple times (Figure 8). Compared with several representative VideoLLMs, our model triggers at more accurate times.

Narration In the narration task, the model must choose when to speak during a long streaming video and provide concise summaries of events observed so far without access to future content. As shown in Figure 9, compared with VideoLLMs, our outputs are more succinct and our response timings align more closely with key event boundaries, leading to more accurate online narration.

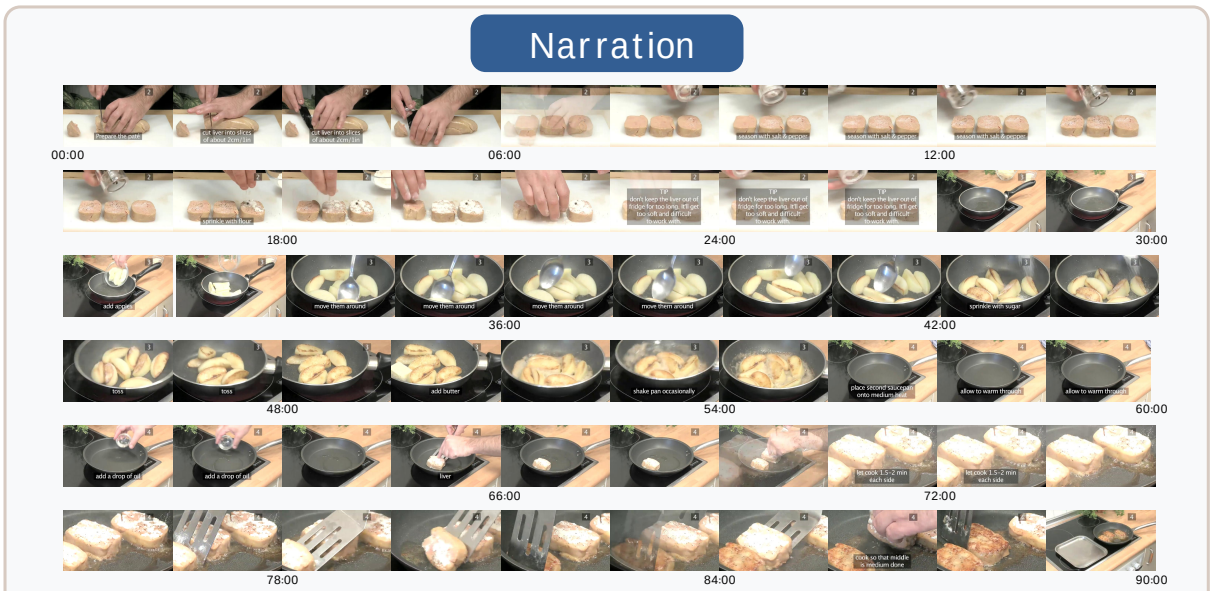
Reactive QA With audio queries in the reactive QA setting (Figure 10), ROMA correctly localizes the relevant segment in the long video and extracts the key visual evidence. In contrast, MiniCPM-o misidentifies the segment, while Qwen2.5-Omni often responds with unnecessary follow-up questions.

A.7 Evaluation Prompt

LLM-as-a-judge is a widely adopted paradigm for scalable evaluation, given its strong alignment with human preferences (Zheng et al., 2023). Accordingly, we employ GPT-4o as a reliable scorer for our open-ended tasks. Detailed prompts are provided in Figures 11 and 12.

Figure 7: Qualitative comparison on the single-alert proactive task. While MMDuet and VideoLLM-Online exhibit premature triggering and hallucination before the target event appears, ROMA accurately accumulates visual evidence to release a precise alert at 17.0s, aligning with the ground truth interval (12s–28s).

Figure 8: Qualitative comparison on the recurring-alert task. While MMDuet suffers from continuous over-generation without distinguishing event boundaries, ROMA effectively tracks the repetitive action, releasing distinct alerts at 7.0s and 15.0s to capture the recurring instances.



Question: Describe various activity events occurring in the video in real time.

Ground Truth:
6.0s | cut the goose liver into small slices
12.0s | season them with salt and pepper
22.0s | sprinkle with flour on both sides
30.0s | heat the apple in a pan with some oil
48.0s | sprinkle some salt on top of the apples and add some butter
81.0s | cook the liver on another pan with oil

ROMA (Ours)

7.0s | Cut the liver into slices.
13.0s | Season the liver with salt and pepper.
24.0s | Sprinkle flour on both sides of the liver.
42.0s | Add apples to the pan and move them around.
59.0s | Add butter to the pan and shake the pan occasionally.
71.0s | Place the liver in the pan and cook until browned.

MMDuet

5.0s | A person is cutting a piece of meat into slices on a cutting board.
11.0s | sprinkle salt and pepper
16.0s | sprinkle flour on the meat
21.0s | sprinkle flour on the meat
27.0s | add seasoning and flour to the meat
34.0s | pour oil into the pan
40.0s | fry meat
46.0s | fry the meat
52.0s | fry the meat
58.0s | add butter
64.0s | fry the meat
70.0s | add the ingredients to the pan
83.0s | put the liver into the pan
87.0s | fry meat

VideoLLM-Online

0.0s | (Video Time = 0s) Assistant: You cut the meat with a knife.

An example from YouCook2

Figure 9: Qualitative comparison on the real-time narration task. While MMDuet suffers from severe repetition and redundant over-generation, ROMA effectively tracks the procedural evolution, generating concise, time-aligned descriptions that correspond strictly to the distinct ground truth events.

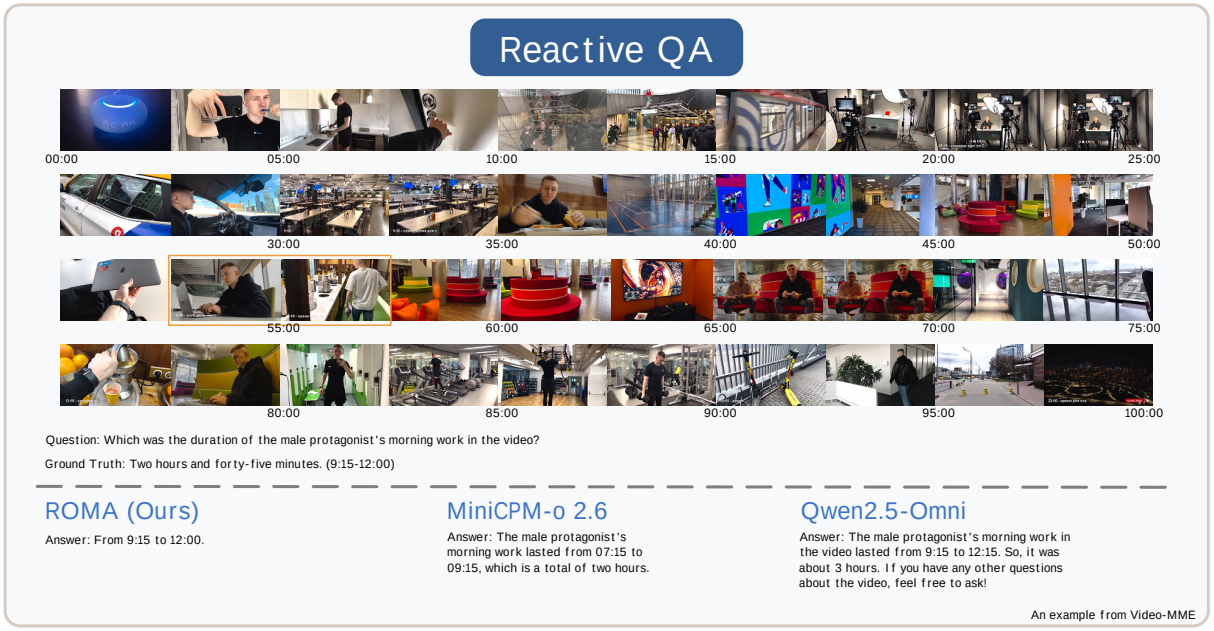


Figure 10: Qualitative comparison on the reactive QA task. While baseline models suffer from temporal misalignment or hallucinated intervals when querying specific activity durations, ROMA accurately retrieves the exact start and end timestamps (9:15–12:00) to derive the correct answer.

Evaluation Prompt for Open-Ended Audio-Query Assessment

```
"""
You are an AI assistant tasked with evaluating whether a response matches the correct answer to
    a given question.

## Evaluation Rules
(1) Output 1 if the response matches the answer exactly or with synonymous/ equivalent wording.
- Synonyms, paraphrases, or different surface forms of the same meaning count as matches.
- Responses containing the correct answer with irrelevant details count as matches.
- Responses providing sufficient information to infer the correct answer count as matches.

(2) Output 0 if the response is incorrect, contradictory, or completely irrelevant to the
    question.
- If the answer and the response address different topics, or if the response does not answer
    the question.
- If the response introduces additional details that change the meaning of the answer, mark as 0.


### Examples
Example 1:
Question: What is the genre of this video?
Answer: It is a news report that introduces the history behind Christmas decorations.
Response: It's a Christmas-themed video, filled with festive decorations and a warm, cozy
    atmosphere. It really captures that classic holiday spirit. What do you think?
Your output: 0

Example 2:
Question: How many birds are above the fireplace?
Answer: 2.
Response: One bird is above the fireplace, and another is below.
Your output: 1

Your Turn:
Question: {question}
Answer: {ground_truth}
Response: {prediction}
Your output:
"""
```

Figure 11: Full prompt provided to GPT-4o for open-ended evaluation with audio queries on Video-MME and EgoSchema.

Evaluation Prompt for Narration Task Evaluation

```
"""
You are an expert evaluator for video narration quality. Your task is to compare
a reference description of a video (ground truth) with a model-generated description
for the same video, and output THREE scores between 0 and 1.

You must consider the model response as a SINGLE long story (it may contain multiple
sentences describing different moments in the video).

IMPORTANT: Higher scores are always better.

Definitions:

1. coherence (story coherence):
   - How internally coherent and well-structured is the model-generated story by itself?
   - Does it read like a reasonable, temporally plausible sequence of actions and states?
   - Penalize contradictions, abrupt jumps, and incoherent, rambling structure.
   - 1.0 = very coherent and well-structured; 0.0 = completely incoherent.

2. alignment (semantic alignment with ground truth):
   - How well does the model-generated story capture the key actions and steps in the ground
     truth?
   - Consider whether important actions/events are present, correctly described, and roughly in
     a reasonable order.
   - Hallucinated major steps that clearly do not appear in the ground truth should reduce this
     score.
   - 1.0 = almost all key content in GT is covered with correct semantics; 0.0 = almost
     completely unrelated.

3. conciseness (relevant non-redundancy / brevity):
   - This score measures whether the model response is concise GIVEN IT IS RELEVANT to the
     ground truth.
   - If the model response is largely unrelated to the ground truth (low semantic overlap, wrong
      topic, ignores the video), conciseness MUST be near 0, even if the response is short.
   - Penalize heavy repetition of similar sentences, long irrelevant digressions, and obvious
     padding.
   - However, do NOT penalize necessary detail that genuinely helps describe the steps.
   - 1.0 = succinct, minimal redundancy while preserving essential details;
     0.0 = extremely repetitive / rambling / full of irrelevant filler / irrelevant with the
     groundtruth.


Empty or meaningless model responses (or responses that ignore the task) should receive low
     scores,
typically near 0 for all dimensions.

Output format (VERY IMPORTANT):
- You MUST output valid JSON with exactly the following keys:
  {"coherence": <float>, "alignment": <float>, "conciseness": <float>}
- Each value must be a number between 0 and 1 (inclusive).
- Do NOT output any extra text or explanation.
"""
```

Figure 12: Prompt used to instruct GPT-4o to evaluate narration quality along three criteria: story coherence, alignment with ground truth, and conciseness.