PROACT-VL2: REINFORCEMENT LEARNING FOR OPEN-ENDED PROACTIVE REAL-TIME STREAMING INTERACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

As AI companions increasingly participate in shared interactive experiences, such as co-watching and commenting on a game video with a user, they must continuously process streaming inputs to determine when and how to contribute. However, existing proactive real-time streaming models primarily optimize for predefined targets via supervised fine-tuning, such as flagging specific events or answering set queries, which falls short in open-ended scenarios, as it fails to ensure that the interaction trajectory remains contextually grounded, dynamically adapts to shifting conversational dynamics, and stays well-coordinated with others. We address this gap with reinforcement learning using a reward that assesses contextual grounding, adaptability, and human-AI coordination. Experiments demonstrate that this approach outperforms supervised learning and reference-based rewards, substantially improving proactive interaction quality and reducing multimodal grounding errors while preserving general video understanding capabilities.

1 INTRODUCTION

AI companions that participate in ongoing human activities must do more than respond to explicit requests. Whether commenting on a game video, guiding a user through a recipe, or collaborating during a presentation, they must continuously follow events, determine when a contribution would be useful, and coordinate with others, including remaining silent when appropriate. Recent real-time video multimodal large language models (video MLLMs) have begun to support these capabilities by processing streaming visual inputs and proactively deciding when and what to say (Chen et al., 2024; Wang et al., 2025c; Qian et al., 2025; Wang et al., 2025a; Lu et al., 2026a; Yan et al., 2026; Wang et al., 2026).

Existing proactive systems often focus on predefined targets, such as detecting a safety hazard or answering a question about a particular object or moment (Wang et al., 2025c;b; Zhao et al., 2026; Niu et al., 2025). Open-ended companionship introduces a different requirement: the same context can admit multiple appropriate choices of whether, when, and what to say. For example, after a scoring play, a game commentator could highlight the decisive pass or explain the defensive mistake that enabled it. Either contribution may be useful, and neither is required for the interaction to succeed. If a partner is already explaining the play, the model could instead wait and add a complementary observation, or remain silent. These alternatives can yield different but valid interaction trajectories. Their quality depends on how contributions fit the unfolding activity and conversation, making agreement with a single reference trajectory an incomplete training objective.

Prior work has used supervised fine-tuning (SFT) to teach video MLLMs when and what to say in proactive streaming interaction (Yan et al., 2026; Wang et al., 2025c). Proact-VL takes an important step toward open-ended companionship by applying SFT to diverse interaction scenarios (Yan et al., 2026). However, broader scenario coverage does not by itself resolve the mismatch between demonstration matching and interaction quality. First, an SFT objective increases the likelihood of the demonstrated behavior without explicitly crediting alternatives that may be equally appropriate. Second, matching demonstrated responses at individual decision points does not directly optimize the quality of the resulting interaction over time. Even with access to past context, a model receives no explicit trajectory-level feedback on whether its contributions become repetitive, fail to adapt

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

to new events, or conflict with its partner's ongoing explanation. Open-ended interaction therefore calls for an objective that accommodates multiple valid behaviors and evaluates their temporal relationships.

We formulate this problem as *trajectory-level behavior alignment* and optimize it with reinforcement learning (RL). RL provides a mechanism for learning from the quality of model-generated interactions without requiring a target response for every decision. Our reward design addresses the two objective mismatches directly. To accommodate valid alternatives, it evaluates behavior against the evolving multimodal and conversational context, assessing visual grounding, adaptation to changing events, and coordination with other speakers rather than agreement with a single reference trajectory. To capture dependencies across decisions, it scores the complete interaction over a video span. This allows a contribution to be evaluated in relation to what the model and its partner have already said and how the interaction subsequently unfolds. Optimizing this trajectory-level reward thus provides feedback on the joint choices of whether, when, and what to contribute.

Experiments show that our approach improves open-ended interaction quality over SFT and RL with reference-based proactive rewards. Behavioral analysis further identifies more varied responses and fewer multimodal grounding and interaction-state tracking errors, while general video understanding capabilities are preserved. These findings connect the gains to the motivating objective gap: improvements extend to how the model responds to visual events and maintains interaction context, beyond agreement with reference responses.

Our contributions are threefold: (1) We formulate open-ended proactive streaming interaction as trajectory-level behavior alignment, capturing both the multiplicity of valid behaviors and the dependencies among decisions over time. (2) We introduce an RL approach with a context-based reward that evaluates complete interaction trajectories for grounding, adaptation, and coordination without requiring agreement with a single reference trajectory. (3) We empirically demonstrate improved proactive interaction quality and fewer grounding and interaction-state tracking errors compared with supervised fine-tuning, while preserving general video understanding.

2 Related Work

2.1 Real-Time Streaming Multimodal Models

Video MLLMs are increasingly moving from offline question answering over bounded videos toward real-time streaming interaction, where models continuously process incoming observations and autonomously determine when to respond (Chen et al., 2024; Huang et al., 2025). Existing approaches broadly realize this capability in two ways. Some augment conventional video MLLMs with explicit response-decision modules: Dispider (Qian et al., 2025) disentangles streaming perception, decision, and response, StreamBridge (Wang et al., 2025a) introduces a lightweight activation model, and ROMA (Xueyun et al., 2026) uses a dedicated speak head to trigger responses. Others directly model real-time interaction within the streaming sequence, enabling the model itself to decide whether to remain silent or respond as new multimodal inputs arrive, as in MMDuet (Wang et al., 2025c) and MiniCPM-o 4.5 (Cui et al., 2026). Training in this line has predominantly relied on supervised learning from temporally aligned streaming interactions, while recent works such as MMDuet2 (Wang et al., 2026) further introduce reinforcement learning to optimize response timing and streaming behavior. Despite these advances, while existing training objectives largely focus on detecting and responding to predefined target events, objects, or response opportunities, our work in contrast optimizes open-ended interaction behavior by trajectory-level reinforcement learning.

2.2 Benchmarks for Real-Time Multimodal Interaction

Evaluation has similarly evolved from offline video QA toward benchmarks that explicitly test streaming and proactive capabilities. Recent streaming benchmarks assess online perception, temporal reasoning, memory, and continuously updated understanding over unfolding video (Lin et al., 2026; Shi et al., 2026; Huang et al., 2025; Yang et al., 2025; Niu et al., 2025), while proactive and interactive benchmarks further evaluate response timing, event triggering, turn-taking, monitoring, and decisions about when to respond or remain silent (Zhao et al., 2026; Wang et al., 2025d; Pourreza et al., 2026; Wang et al., 2025b; Lu et al., 2026b). However, most existing evaluations still

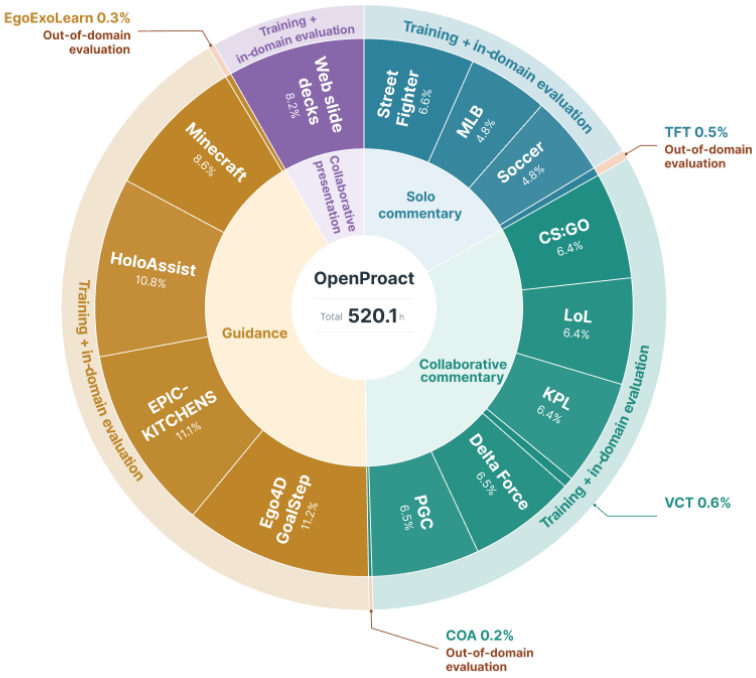


Figure 1: Data sources and duration distribution of OpenProact. The inner ring groups sources by interaction setting, the middle ring presents each source's share of aggregate duration, and the outer ring indicates training/in-domain vs. out-of-domain use.

organize interaction around predefined questions, standing requests, target events, or annotated response opportunities, whereas our work evaluates whether a model can continuously choose useful, context-dependent behavior over complete open-ended interaction trajectories.

3 OpenProact: Data for Open-Ended Proactive Interaction

Open-ended proactive interaction requires learning not only what to say, but when to respond, intervene, and coordinate with others. Building on Proact-VL's settings (Yan et al., 2026), **OpenProact** broadens supervision across three axes: domains (games, sports, and everyday activities), communicative functions (event narration, anticipatory task guidance, and joint presentation), and interaction structures (single-speaker and partner-conditioned), enabling the study of when and what to say under diverse temporal, social, and discourse constraints.

3.1 Data Sources and Construction

We construct **OpenProact** across four representative interaction settings: solo commentary, collaborative commentary, guidance, and collaborative presentation. Figure 1 summarizes the data distribution. Detailed construction procedures are provided in Appendix C.1.

Solo and collaborative commentary. We collect naturally occurring game and sports broadcasts from Bilibili, including CS:GO, League of Legends, Valorant, and soccer. We recover timestamped speaker turns with ASR and speaker diarization, and use a language model to clean the transcripts and extract speaker-specific personas. Single-speaker broadcasts provide solo commentary, while multi-speaker broadcasts provide collaborative commentary with temporally aligned partner speech.

Guidance. We construct guidance interactions from both real-life tasks and game environments. For real-life guidance, we use Ego4D GoalStep, EPIC-KITCHENS, and HoloAssist (Grauman et al.,

2022; Damen et al., 2021; Wang et al., 2023), converting annotated goals and action steps into concise task-specific guidance. For Minecraft, a vision-language model identifies goal-directed intervals and corresponding actions from instructional gameplay videos. In both cases, each instruction is aligned to occur before the action it is intended to support, producing proactive guidance rather than post-hoc narration.

Collaborative presentation. Because naturally occurring, temporally aligned two-presenter data are scarce, we collect presentation slide decks and synthesize slide-grounded dialogues between two presenter roles. A language model generates role-consistent turns conditioned on the slide content and preceding dialogue, and synthesized speech durations determine turn boundaries and temporal alignment.

3.2 Data Splits

We organize OpenProact into four splits: (1) OpenProact-SFT: 39,156 clips totaling 461.3 hours, used for supervised fine-tuning. (2) OpenProact-RL: 1,992 interaction trajectories totaling 23.7 hours, used for reinforcement learning. (3) OpenProact-ID: held-out examples from the same source domains as the training data, used for in-domain evaluation. (4) OpenProact-OOD: data from TFT, COA, and EgoExoLearn, all excluded from both SFT and RL training, used for out-of-domain evaluation. Detailed source-level statistics and split procedures are provided in Appendix C.2.

4 Training

4.1 Supervised Fine-Tuning for Proactive Streaming Interaction

We first use supervised fine-tuning (SFT) to establish a base policy for learning *when and what to say* in proactive streaming interaction. SFT optimizes the model output at each one-second decision step conditioned on all previously observed interaction context, with the target being either a textual utterance or an explicit [SILENT] action. To maintain a realistic streaming speech rate, each speaking step is limited to at most six words, while longer utterances can naturally continue across consecutive steps.

Based on our preliminary experiments, we adopt a weighted SFT objective to mitigate the model's tendency to over-predict [SILENT]. Although speaking and silent steps are approximately balanced in the training data, uniform supervision gradually biases the model toward silence. We find that this is largely driven by repeated idle silence, while transitions between speaking and silence are more difficult to learn and more important for interaction behavior. We therefore down-weight repeated idle silence to 0.12 and assign partner-deferral silence a weight of 0.35. The first silent step following model speech receives weight 1.0 to preserve the boundary of a speaking turn. Conversely, the first speaking step after silence is up-weighted to 1.5 to encourage speech initiation, followed by weights 1.3 and 1.15 for the next two speaking steps to stabilize speech continuation. All remaining assistant outputs use unit weight. This weighting scheme provides a more balanced initialization of speaking, silence, and their transitions before reinforcement learning.

4.2 Trajectory-Level Reinforcement Learning for Open-ended Streaming Interaction

While SFT provides a useful initialization for learning when and what to say, its frame-wise imitation objective is limited for open-ended streaming interaction. In particular, it (1) encourages the model to reproduce a single demonstrated trajectory even when multiple interaction behaviors may be equally appropriate, and (2) optimizes individual frame-level responses without directly accounting for the quality and coordination of the interaction as it unfolds over time. Our trajectory-level reinforcement learning formulation addresses these limitations through two components. (1) We *roll out the policy causally over the full interaction*, so that each decision becomes part of the context for subsequent decisions and the model is optimized based on the resulting interaction trajectory rather than independently matching a target at each frame. (2) We *evaluate the completed trajectory with an open-ended multimodal reward*, which assesses whether the interaction is appropriate under the evolving visual and conversational context without requiring the model to reproduce a sin-

gle demonstrated behavior. Together, these components enable optimization of how speaking and silence decisions unfold, interact, and remain coordinated over time

4.2.1 TRAJECTORY-LEVEL ROLLOUT

We roll out the policy causally over the full interaction, allowing each model decision to influence the context of subsequent decisions. Starting from the first step, the model observes the interaction history available up to the current time, produces an output y_t, and appends this output to the history before proceeding to the next step. This process is repeated for all T steps.

Formally, for an interaction over T causally revealed video chunks $V_{1:T} = (V_1, \ldots, V_T)$, the policy at step t is

$$y_t \sim \pi_\theta \left(\cdot \mid V_{\leq t}, U_{<t}^{\text{partner}}, y_{<t}, c \right), \qquad y_t \in \mathcal{Y} \cup [\text{SILENT}], \tag{1}$$

where $V_{\leq t}$ denotes the visual stream observed so far, $y_{<t}$ contains the model's previous outputs, and c denotes static context such as the task instruction and assigned persona. In collaborative commentary and collaborative presentation, $U_{<t}^{\text{partner}}$ denotes the completed utterances from the other speaker in the interaction; it is absent in single-speaker scenarios. To reflect real-world interaction, the model can only condition on partner speech that has been completed before the current step. Any partner speech produced during step t becomes available only at step $t + 1$, which prevents the model from accessing information that would not yet be observable in a causal interaction.

Each sampled output is incorporated into the context for future decisions. Repeating this causal rollout yields the complete interaction trajectory

$$\tau = (y_1, \ldots, y_T), \tag{2}$$

which is subsequently evaluated as a whole by the trajectory-level reward.

4.2.2 TRAJECTORY-LEVEL REWARD

Multimodal Trajectory Quality. As interaction quality depends on how content, timing, and coordination unfold over the entire trajectory rather than at individual steps, we introduce **Multimodal Trajectory Quality (MM-TrajQ)** as our primary semantic reward, which evaluates the quality of a completed trajectory of streaming interaction. The judge is given the corresponding video segment, the temporally aligned partner speech when applicable, the assigned role or persona, and the complete model trajectory. It evaluates whether the interaction remains visually grounded and contextually appropriate over time, including response timing, specificity, continuity across consecutive decisions, usefulness of the contributions, and coordination with the interaction partner.

Length Penalty. During preliminary RL experiments, we observed that the model could obtain high semantic reward by producing unrealistically long utterances within a single streaming step. We therefore introduce a simple length penalty $P_{\text{len}}(\tau)$ to constrain the amount of speech produced within each one-second step. We allow up to six words for each non-[SILENT] output; words beyond this budget are penalized proportionally, with every additional 10 words contributing a penalty of 1, capped at 1.

For trajectory τ_i, we define its MM-TrajQ reward as $R_{\text{MTQ}}(\tau_i) = s_i$. Let $P_{\text{len}}(\tau_i)$ denote the length penalty. We set $w_{\text{MTQ}} = 2$ and $\lambda_{\text{len}} = 1$. The final reward is

$$R(\tau_i) = w_{\text{MTQ}} R_{\text{MTQ}}(\tau_i) - \lambda_{\text{len}} P_{\text{len}}(\tau_i). \tag{3}$$

The complete MM-TrajQ evaluation rubric and exact implementation of the length penalty are provided in Appendix A.3.

4.3 TRAINING DETAILS

We initialize SFT from Qwen3-VL-4B and train on OpenProact-SFT, which contains approximately 40K data. Reinforcement learning starts from the resulting SFT checkpoint and uses OpenProact-RL, which contains approximately 2K data. For each RL trajectory, we sample a 20–50 second span from the underlying video. This duration is long enough to contain multiple speaking opportunities and changes in the visual or conversational context, while keeping rollout and multimodal evaluation

Table 1: Results on the in-domain evaluation set. We report proactivity F1 and three interaction-quality metrics: LiveU, FinalQ (FQ), and MM-TrajQ (MT.) for each interaction setting and overall. The best and second-best results are shown in **bold** and underline, respectively. Model size denotes the parameter scale of the evaluated model or backbone; "–" indicates that the size is not publicly disclosed.

		Solo Comm.				Collab. Comm.				Guidance				Collab. Pres.				Overall			
Model	**Size**	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.
StreamingVLM	7B	.578	3.21	3.65	2.51	.625	2.76	3.41	2.27	.617	2.51	2.97	1.99	.783	2.58	3.38	3.14	.655	2.77	3.35	2.47
LiveCC	7B	**.690**	3.43	4.56	3.00	.706	3.09	4.52	2.92	.580	3.46	4.96	2.71	.592	5.65	6.17	4.52	.652	3.91	5.05	3.29
MMDuet2	3B	.271	3.14	3.70	2.80	.241	2.66	3.86	2.46	.034	2.43	2.20	2.07	.013	2.48	5.43	2.88	.159	2.68	3.80	2.55
Mage-VL	5B	.160	2.10	3.36	1.92	.158	3.15	3.88	2.45	.148	3.51	3.88	3.14	.088	3.61	4.76	3.71	.137	3.09	3.97	2.81
Proact-VL	7B	.678	4.36	4.31	2.98	.791	5.66	4.76	3.77	.455	**7.22**	**5.54**	4.29	.581	7.57	6.43	**6.81**	.660	6.20	5.26	4.46
Gemini Live	–	.465	**5.80**	2.24	**4.61**	.661	5.18	**5.15**	3.10	.308	5.86	4.02	**5.00**	.405	6.58	5.94	5.80	.498	5.86	4.34	4.63
Proact-VL2$_{\text{SFT}}$	4B	.672	4.89	3.66	2.37	.807	5.77	4.27	3.26	**.638**	5.74	4.62	3.25	**.982**	7.84	6.52	6.29	**.784**	6.06	4.77	3.79
Proact-VL2$_{\text{RL}}$	4B	.680	4.94	**4.59**	4.05	**.824**	**5.82**	4.97	**4.76**	.525	6.59	5.27	4.05	.978	**7.93**	**6.63**	6.48	.780	**6.32**	**5.36**	**4.84**

costs manageable. We optimize the reward in Equation 3 with GRPO using four causal rollouts per context; rewards are standardized within each group, and all tokens in a rollout share its group-relative advantage. MM-TrajQ is computed by GPT-5.2. Both SFT and RL are trained on $8\times$ NVIDIA H100 80GB GPUs. SFT takes approximately 7 hours, while RL takes approximately 30 hours. Additional optimization, rollout, and system configurations are provided in Appendix A.

5 Experiments

5.1 Experimental Setup

Evaluation datasets. We evaluate the model on three types of data: (1) in-domain evaluation on OpenProact-ID, which contains held-out examples from the same source domains as training; (2) out-of-domain evaluation on OpenProact-OOD, which contains unseen source domains excluded from both SFT and RL training; and (3) general video understanding on MVBench and Video-MME, to assess whether optimizing proactive interaction preserves the model's general video understanding capability. Detailed evaluation splits and protocols are provided in Appendix B.

Baselines. We compare with representative proactive real-time streaming models, including MMDuet2 (Wang et al., 2026), StreamingVLM (Xu et al., 2026), LiveCC (Chen et al., 2025), Mage-VL (Yang et al., 2026), Proact-VL (Yan et al., 2026), and Gemini Live (Google, 2026), covering both open- and closed-source proactive real-time streaming models.

Evaluation metrics. We use complementary metrics that capture different aspects of proactive interaction. Following prior work (Zhao et al., 2026; Yan et al., 2026), F1 (↑) measures proactivity by comparing the model's speaking decisions with annotated response windows. Following Proact-VL (Yan et al., 2026), LiveU (↑) evaluates second-level streaming usability in terms of timing and turn-taking, speech rate and pacing, and the clarity of individual spoken units, whereas FinalQ (↑) evaluates the fidelity, continuity, and substance of the concatenated spoken content. Proact-VL validated both metrics against human judgments on 100 sampled instances, reporting strict pairwise preference accuracies of 84.7–100.0% for LiveU and 74.0–96.8% for FinalQ across different model pairs. MM-TrajQ (↑) evaluates the complete temporally aligned multimodal interaction trajectory and also serves as our semantic RL reward. We separately validate MM-TrajQ in a model-blinded human study of Proact-VL, Proact-VL2$_{\text{SFT}}$, and Proact-VL2$_{\text{RL}}$ on the same 20 clips, comprising 60 clip–model pairs and 180 ratings. Across all three systems, MM-TrajQ shows strong alignment with aggregated human ratings (Spearman's $\rho = 0.917$; linearly weighted $\kappa = 0.878$). Metric definitions are provided in Appendix B.3, and our human annotation protocol is detailed in Appendix D.

5.2 Main Results

Table 1 shows that Proact-VL2$_{\text{RL}}$ obtains the best overall LiveU, FinalQ, and MM-TrajQ, and an F1 of .780. It substantially outperforms comparable-scale baselines such as MMDuet2 (3B) and Mage-

Table 2: Comparison of reward formulations. We report proactivity F1 and three interaction-quality metrics: LiveU, FinalQ (FQ), and MM-TrajQ (MT.). The best and second-best results are shown in **bold** and underline, respectively.

	Solo				Collab. Comm.				Guidance				Collab. Pres.				Overall			
Reward	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.	F1	LiveU	FQ	MT.
Original PAUC	.006	1.47	2.46	1.41	.006	1.45	3.66	1.47	.098	2.58	2.00	1.96	.933	7.89	6.57	6.35	.427	3.35	3.67	2.80
MMDuet2 PAUC	.464	**6.46**	3.90	2.81	.530	**7.01**	4.44	3.71	**.620**	5.84	4.37	3.37	.963	7.62	5.33	**6.49**	.674	**6.73**	4.51	4.09
Ours	**.680**	4.94	**4.59**	**4.05**	**.824**	5.82	**4.97**	**4.76**	.525	**6.59**	**5.27**	**4.05**	**.978**	**7.93**	**6.63**	6.48	**.780**	6.32	**5.36**	**4.84**

Table 3: Ablation of RL reward.

Reward	**MM-TrajQ** ↑	**Overlength (%)** ↓
$2R_{\mathrm{MTQ}}$	4.74	1.13
$2R_{\mathrm{MTQ}} - P_{\mathrm{len}}$	**4.84** (+0.10)	**0.03** (−97.3%)

VL (5B), and also surpasses larger 7B models including StreamingVLM, LiveCC, and Proact-VL across all four overall metrics. Moreover, RL substantially improves interaction quality over the SFT model, increasing overall LiveU from 6.06 to 6.32, FinalQ from 4.77 to 5.36, and MM-TrajQ from 3.79 to 4.84, while maintaining comparable overall proactivity performance (F1: .784 vs. .780). These results demonstrate the effectiveness of RL for improving interaction quality without substantially sacrificing overall proactive behavior.

5.3 Analysis

Trajectory-Level Reward vs. Reference-Based Rewards. A key question is whether the gains from RL come from reinforcement learning itself or from the design of the reward used to optimize the interaction trajectory. We therefore compare our reward against prior rewards for proactive video interaction. To our knowledge, MMDuet2 is the only prior proactive video MLLM that applies reinforcement learning (Wang et al., 2026); we compare against both the original PAUC reward and the refined PAUC reward adopted in MMDuet2. Both PAUC variants reward temporal agreement between the model's speaking behavior and a ground-truth response trajectory, whereas our reward directly evaluates the quality of the resulting multimodal interaction trajectory. All three variants use identical RL settings and the same penalty term proposed in this work, differing only in the reward component. As shown in Table 2, compared with the original PAUC and MMDuet2 PAUC rewards, respectively, our reward increases overall F1 from .427/.674 to .780, FinalQ from 3.67/4.51 to 5.36, and MM-TrajQ from 2.80/4.09 to 4.84. It achieves the best overall F1, FinalQ, and MM-TrajQ and improves interaction quality across the four settings. These gains across diverse interaction scenarios and complementary metrics suggest that directly optimizing trajectory quality is more effective for open-ended proactive interaction than matching a single annotated response trajectory.

Effect of the Length Penalty. We ablate the length penalty in our RL objective. Optimizing MM-TrajQ alone substantially improves interaction quality but leads to frequent overlength outputs, where the model produces an unrealistically large number of words within a single one-second streaming step. As shown in Table 3, although the overlength rate is only 1.13%, this still corresponds to more than one such abrupt output every 100 seconds of interaction, making the resulting behavior noticeably unrealistic. Adding P_{len} reduces the overlength rate to 0.03%, nearly eliminating these violations, while further improving MM-TrajQ from 4.74 to 4.84.

Human evaluation. Human ratings further support the interaction-quality gains from RL. In the paired SFT–RL analysis of our three-model study, the mean human score increases from 3.87 for Proact-VL2$_{\mathrm{SFT}}$ to 6.18 for Proact-VL2$_{\mathrm{RL}}$ on a 1–10 scale, with a mean paired improvement of 2.32 points (95% paired-bootstrap CI $[1.63, 3.02]$). After averaging the three ratings for each clip–model pair, RL scores higher than SFT on 19 of the 20 shared clips. These results corroborate the automated evaluation with direct human assessments of interaction quality. Full results and the annotation protocol are provided in Appendix D.

Table 4: Performance on held-out OOD evaluation domains.

Method	OOD Domain	F1↑	LiveU↑	FinalQ↑	MM-TrajQ↑
Proact-VL2$_{SFT}$	**Overall**	0.7671	5.75	4.15	2.01
	TFT	0.8181	**6.00**	4.05	1.70
	COA	0.8288	5.67	3.74	1.81
	EgoExoLearn	**0.6233**	5.53	4.41	2.40
Proact-VL2$_{RL}$	**Overall**	**0.7953**	5.70	**5.48**	**5.11**
	TFT	**0.8551**	5.69	**5.46**	**5.46**
	COA	**0.8644**	5.51	**5.31**	**5.33**
	EgoExoLearn	0.6005	**5.79**	**5.56**	**4.66**

Table 5: General video understanding on established benchmarks.

Model	MVBench↑	Video-MME↑
Qwen3-VL	63.28	56.0
MMDuet2	62.08	52.1
Proact-VL2$_{SFT}$	61.40	52.4
Proact-VL2$_{RL}$	61.88	54.7

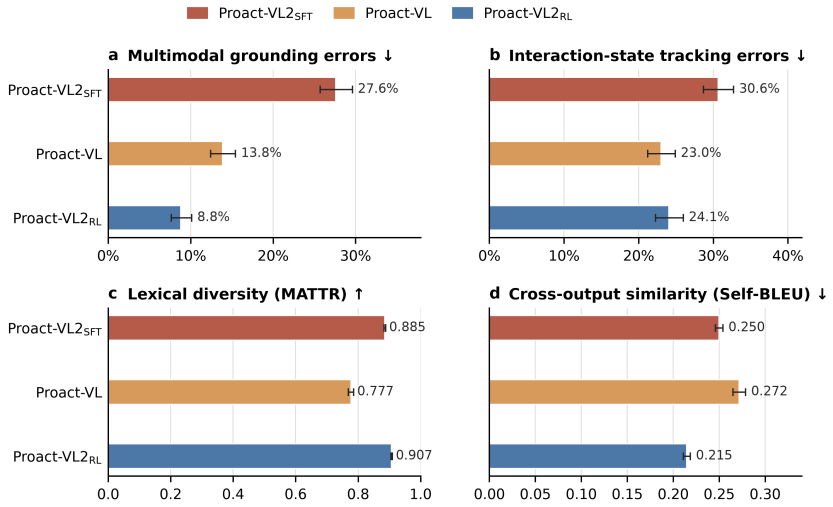


Figure 2: **Behavioral analysis of proactive interaction policies. (a–b)** Frequency of multimodal grounding and interaction-state tracking errors (lower is better). **(c–d)** Response diversity measured by MATTR (higher is better) and Self-BLEU (lower is better).

5.4 Generalization

Performance on OOD Data. Table 4 shows that Proact-VL2$_{RL}$ generalizes beyond the training domains, improving overall FinalQ from 4.15 to 5.48 and MM-TrajQ from 2.01 to 5.11 over Proact-VL2$_{SFT}$, with overall F1 of .7953 versus .7671.

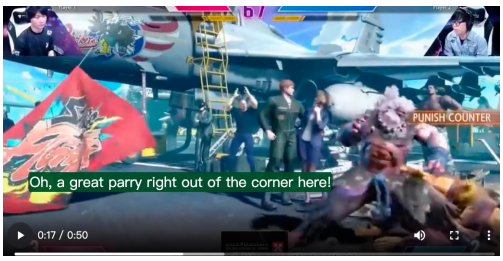
Preserving general video understanding. Table 5 shows that Proact-VL2$_{RL}$ largely preserves the base model's general video understanding, with only modest drops on MVBench (1.40) and Video-MME (1.3). RL also partially recovers the degradation introduced by SFT, improving by 0.48 and 2.3 points on the two benchmarks, respectively, while remaining competitive with MMDuet2.

5.5 Behavioral Analysis.

Beyond aggregate interaction scores, we examine how RL changes the model's behavior along two complementary dimensions: interaction reliability and output diversity. Figure 2 reports quantitative diagnostics, while Figure 3 provides representative examples of the observed failure modes.

Failure patterns. We identify two recurring errors: *multimodal grounding errors*, where the response contains claims unsupported by the current observation, and *interaction-state tracking errors*, where the response relies on an outdated interaction state. Relative to the Proact-VL2$_{SFT}$, Proact-VL2$_{RL}$ reduces grounding errors from 27.6% to 8.8% and state-tracking errors from 30.6% to 24.1%. Compared with Proact-VL, its grounding-error rate is lower (8.8% vs. 13.8%), while its state-tracking error rate is 24.1% vs. 23.0%. The examples in Figure 3 illustrate these patterns: in Street Fighter, Proact-VL2$_{SFT}$ incorrectly describes an unsupported parry, whereas RL

Multimodal Grounding Error



(a) Street Fighter — Proact-VL2$_{\text{SFT}}$



(b) Street Fighter — Proact-VL2$_{\text{RL}}$

Interaction-State Tracking Error



(c) Minecraft — Proact-VL2$_{\text{SFT}}$



(d) Minecraft — Proact-VL2$_{\text{RL}}$

Figure 3: **Representative failure cases before and after RL.**

remains grounded in the observed offensive exchange; in Minecraft, Proact-VL2$_{\text{SFT}}$ continues giving swimming-related guidance after the interaction has progressed toward the shore, while RL updates its response to the current state. The annotation protocol is provided in Appendix E.

Output diversity. We evaluate response diversity using two widely adopted metrics: MATTR for lexical diversity (Baccini & Urpelainen, 2015) and Self-BLEU for cross-output similarity (Zhu et al., 2018). Proact-VL2$_{\text{RL}}$ produces more diverse responses, achieving the highest MATTR (0.907 vs. 0.885 for Proact-VL2$_{\text{SFT}}$ and 0.777 for Proact-VL) and the lowest Self-BLEU (0.215 vs. 0.250 and 0.272, respectively). These results suggest that trajectory-level optimization improves interaction quality without collapsing toward a narrow set of response patterns.

6 Conclusion

In this work, we introduced a novel reinforcement learning framework that formulates open-ended proactive streaming interaction as trajectory-level behavior alignment, moving beyond the limitations of step-wise supervised fine-tuning and predefined target matching. By optimizing complete interaction trajectories using an open-ended multimodal reward, our approach enables AI companions to continuously determine whether, when, and what to communicate based on evolving visual and conversational contexts rather than merely imitating frame-level targets. Extensive experiments on the newly constructed OpenProact dataset demonstrate that this reinforcement learning strategy substantially improves overall interaction quality across diverse scenarios, including solo commentary, collaborative commentary, guidance, and collaborative presentation, yielding more lexically diverse and less self-similar responses than comparable baselines. Furthermore, our method significantly reduces multimodal grounding and interaction-state tracking errors while effectively preserving the underlying model's general video understanding capabilities, marking a critical step toward the development of more natural, adaptive, and reliable real-time AI companions.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data, clean and reformat datasets, support qualitative annotation, assist with writing code, and edit the manuscript to improve readability. We did not use generative AI tools to generate research ideas or propose or refine hypotheses, design or provide feedback on the research methodology or experiments, or interpret results and draw scientific conclusions. We did not use generative AI tools for any other tasks requiring disclosure. All AI-assisted text, synthetic data, processed datasets, and annotations were reviewed by the authors before use, and all AI-assisted code was reviewed and verified by the authors before use. The research ideas, methods, interpretation of results, and scientific conclusions are the authors' own. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide anonymous access to our training code at `https://anonymous.4open.science/r/Proact-VL2-D1C6/` and our dataset construction pipeline at `https://anonymous.4open.science/r/ProactiveBench-77CB/`. The appendix describes the training hyperparameters and implementation details used in our experiments.

REFERENCES

Leonardo Baccini and Johannes Urpelainen. *Cutting the Gordian knot of economic reform: When and how international institutions help*. Oxford University Press, 2015.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large language model for streaming video. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18407–18418, 2024. doi: 10.1109/CVPR52733.2024.01742.

Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. Livecc: Learning video llm with streaming speech transcription at scale. In *CVPR*, 2025.

Junbo Cui, Bokai Xu, Chongyi Wang, Tianyu Yu, Weiyue Sun, Yingjing Xu, Tianran Wang, Zhihui He, Wenshuo Ma, Tianchi Cai, Jiancheng Gui, Luoyuan Zhang, Xian Sun, Fuwei Huang, Moye Chen, Zhuo Lin, Hanyu Liu, Qingxin Gui, Qingzhe Han, Yuyang Wen, Huiping Liu, Rongkang Wang, Yaqi Zhang, Hongliang Wei, Chi Chen, You Li, Kechen Fang, Jie Zhou, Yuxuan Li, Guoyang Zeng, Chaojun Xiao, Yankai Lin, Xu Han, Maosong Sun, Zhiyuan Liu, and Yuan Yao. Minicpm-o 4.5: Towards real-time full-duplex omni-modal interaction, 2026. URL `https://arxiv.org/abs/2604.27393`.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2021. doi: 10.1109/TPAMI.2020.2991965.

Google. Introducing Gemini 3.8 Live and 3.8 Live Extended Thinking. https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-8-live-gemini-3-8-live-extended-thinking/, September 2026.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online video understanding: Ovbench and videochat-online. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3328–3338, 2025. doi: 10.1109/CVPR52734.2025.00316.

Junming Lin, Zheng Fang, Chi Chen, Haoxuan Cheng, Zihao Wan, Fuwen Luo, Ziyue Wang, Peng Li, Yang Liu, and Maosong Sun. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12147–12151, 2026. doi: 10.1109/ICASSP55912.2026.11463959.

Xudong Lu, Yang Bo, Jinpeng Chen, Shuhan Li, Xintong Guo, Huankang Guan, Fang Liu, Dunyuan Xu, Peiwen Sun, Heyang Sun, Rui Liu, and Hongsheng Li. Aura: Always-on understanding and real-time assistance via video streams, 2026a. URL https://arxiv.org/abs/2604.04184.

Xudong Lu, Xueying Li, Annan Wang, Yang Bo, Jinpeng Chen, Zengliang Li, Nianzu Yang, Rui Liu, Xue Yang, Jingwen Hou, and Hongsheng Li. Omniinteract: Benchmarking real-world streaming interaction for real-time omnimodal assistants, 2026b. URL https://arxiv.org/abs/2605.26485.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, and Jiaqi Wang. Ovo-bench: How far is your video-llms from real-world online video understanding? In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18902–18913, 2025. doi: 10.1109/CVPR52734.2025.01761.

Reza Pourreza, Rishit Dagli, Apratim Bhattacharyya, Sunny Panchal, Guillaume Berger, and Roland Memisevic. Can vision-language models answer face to face questions in the real-world? In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 70944–70973, 2026.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24045–24055, 2025. doi: 10.1109/CVPR52734.2025.02239.

Yansong Shi, Qingsong Zhao, Tianxiang Jiang, Xiangyu Zeng, Yi Wang, and Limin Wang. River: A real-time interaction benchmark for video llms, 2026. URL https://arxiv.org/abs/2603.03985.

Haibo Wang, Bo Feng, Zhengfeng Lai, Mingze Xu, Shiyu Li, Weifeng Ge, Afshin Dehghan, Meng Cao, and Ping Huang. StreamBridge: Turning your offline video large language model into a proactive streaming assistant. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 132332–132359. Curran Associates, Inc., 2025a. doi: 10.52202/085713-4406. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/bf6939f9058a391c47014731b2486e2a-Paper-Conference.pdf.

Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, et al. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20213–20224. IEEE, 2023.

Yueqian Wang, Xiaojun Meng, Yifan Wang, Huishuai Zhang, and Dongyan Zhao. Proactivevideoqa: A comprehensive benchmark evaluating proactive interactions in video large language models, 2025b. URL https://arxiv.org/abs/2507.09313.

Yueqian Wang, Xiaojun Meng, Yuxuan Wang, Jianxin Liang, Jiansheng Wei, Huishuai Zhang, and Dongyan Zhao. VideoLLM knows when to speak: Enhancing time-sensitive video comprehension with video-text duet interaction format. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 6338–6359, Suzhou, China, November 2025c. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.336. URL https://aclanthology.org/2025.findings-emnlp.336/.

Yueqian Wang, Songxiang Liu, Disong Wang, Nuo Xu, Guanglu Wan, Huishuai Zhang, and Dongyan Zhao. MMDuet2: Enhancing proactive interaction of video mllms with multi-turn reinforcement learning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 23257–23270, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/281970ef377580ab9ec3c7d3781110d4-Paper-Conference.pdf.

Yuxuan Wang, Yueqian Wang, Bo Chen, Tong Wu, Dongyan Zhao, and Zilong Zheng. Omni-immi: A comprehensive multi-modal interaction benchmark in streaming video contexts. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18925–18935, 2025d. doi: 10.1109/CVPR52734.2025.01763.

Ruyi Xu, Guangxuan Xiao, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, and Song Han. Streamingvlm: Real-time understanding for infinite video streams. In *International Conference on Learning Representations*, volume 2026, pp. 61463–61475, 2026.

Tian Xueyun, Wei Li, Bingbing Xu, Heng Dong, Yuanzhuo Wang, and Huawei Shen. ROMA: Real-time omni-multimodal assistant with interactive streaming understanding. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 23018–23039, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1153. URL https://aclanthology.org/2026.findings-acl.1153/.

Weicai Yan, Yuhong Dai, Qi Ran, Haodong Li, Wang Lin, Tao Jin, Xing Xie, Hao Liao, and Jianxun Lian. Proact-VL: A proactive videoLLM for real-time AI companions. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=k9PKgV0L4C.

Senqiao Yang, Kaichen Zhang, Zhaoyang Jia, Jinghao Guo, Yifei Shen, Xinjie Zhang, Xiaoyi Zhang, Haoqing Wang, Xiao Li, Peng Zhang, Xiang An, Yin Xie, Zhening Liu, Xun Guo, Jiahao Li, Shicheng Zheng, Jinglu Wang, Zongyu Guo, Wenxuan Xie, Zihan Zheng, Yuxuan Luo, Bin Li, and Yan Lu. Mage-vl: An efficient codec-native streaming multimodal foundation model, 2026. URL https://arxiv.org/abs/2607.24904.

Zhenyu Yang, Yuhang Hu, Zemin Du, Dizhan Xue, Shengsheng Qian, Jiahong Wu, Fan Yang, Weiming Dong, and Changsheng Xu. SVBench: A benchmark with temporal multi-turn dialogues for streaming video understanding. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 42522–42556, 2025.

Ruixiang Zhao, Jie Yang, Zijie Xin, Tianyi Wang, Fengyun Rao, Jing LYU, and Xirong Li. Omnipro: A comprehensive benchmark for omni-proactive streaming video understanding, 2026. URL https://arxiv.org/abs/2605.18577.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pp. 1097–1100, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3210080. URL https://doi.org/10.1145/3209978.3210080.

A Additional Experimental Details

A.1 Supervised Fine-Tuning

We initialize from Qwen3-VL-4B and train on OpenProact-SFT for two epochs. We apply LoRA to the language-model attention and MLP projections, with rank 64, scaling factor 128, and dropout 0.05. The vision encoder is frozen, while the vision–language projector is trained. Training uses bfloat16, an effective batch size of 64, 5% linear warmup followed by linear decay, zero weight decay, and NEFTune noise with magnitude 5. We use a peak learning rate of 1×10^{-4}, with linear warmup over the first 5% of training steps followed by linear decay to zero. The LoRA weights are merged into the backbone, and the trained projector is retained for RL.

A.2 Causal Rollout and Reinforcement Learning

Rollout construction. Training episodes span 20–50 seconds. The policy receives two frames per second and makes one decision per second, with a maximum of 60 decisions. The frame-area budget ranges from 128×28^2 to 196×28^2 pixels. Generated outputs remain in the dialogue history. Partner speech becomes visible on the following decision step and its original timestamps are retained for overlap measurement.

Reinforcement learning. We use GRPO with four rollouts per context and eight contexts per training batch. Rewards are normalized within each group, and generated tokens share the advantage of their trajectory. Optimization uses AdamW, one PPO epoch per batch, a configured PPO mini-batch size of eight, and a micro-batch size of one per GPU. The optimizer uses $\beta = (0.9, 0.999)$ and the weight decay 0.01. The KL loss coefficient is set to 0.01. The learning rate is set to 1e-7. The entropy coefficient is zero, and the vision tower is frozen. Under this setting, it takes 247 steps to finish training and we choose the checkpoint at the final step as our final RL model.

A.3 Reward Implementation

MM-TrajQ. For each training context, the multimodal judge scores sampled rollouts independently in a single call, using the shared video, task and persona, one reference trajectory, and partner speech when available. Each rollout τ_i receives an integer score $q_i \in \{1, \ldots, 10\}$, which is normalized before reward aggregation. Judge rationales and suggestions are logged for diagnostics and excluded from optimization. The exact judge system prompts, user templates, and output schema are provided in Appendix F.

Length penalty. For output y_t, let $w(y_t)$ denote its whitespace-delimited word count and $e(y_t)$ the number of non-trailing ellipses. A trailing ellipsis is treated as a continuation marker; each remaining ellipsis contributes six effective words. We define

$$\widetilde{w}(y_t) = w(y_t) + 6e(y_t). \tag{4}$$

We accumulate normalized overflow beyond six effective words over speaking steps and clip the trajectory-level total at one:

$$P_{\text{len}}(\tau) = \min\left(1, \frac{1}{10} \sum_{t:m_t=1} \max\left(0, \widetilde{w}(y_t) - 6\right)\right). \tag{5}$$

B Evaluation Details

B.1 Evaluation Datasets

OpenProact-ID. We evaluate in-domain proactive interaction on held-out examples from the same source domains as OpenProact-SFT and OpenProact-RL. The evaluation covers solo commentary, collaborative commentary, guidance, and collaborative presentation.

OpenProact-OOD. We evaluate transfer to unseen source domains using TFT for solo commentary, COA for collaborative commentary, and EgoExoLearn for guidance. These domains are excluded from both SFT and RL training. Source-level construction, counts, and split procedures are provided in Appendix C.

General video understanding. We evaluate MVBench and Video-MME to measure whether proactive interaction training preserves general video understanding.

B.2 Causal Inference Protocol

All models are evaluated causally. At each one-second step t, the adapter conditions on the task and persona, video observed through t, partner speech completed before t, and prior model outputs. The resulting decision is timestamped and appended to the interaction history. Reference responses are excluded from assistant history, and partner speech at step t becomes available at $t + 1$.

For Proact-VL2$_{\text{SFT}}$ and Proact-VL2$_{\text{RL}}$, each step uses two frames by default. Clip-level evaluation retains the full observed prefix, whereas complete-video evaluation uses a 60-second sliding window over visual and dialogue context while preserving the system prompt and task query. Both models use the same frozen evaluation configuration, isolating the effect of training.

Decoding is greedy, with at most 64 new tokens per decision and a repetition penalty of 1.05. Temperature and top-p/top-k sampling are disabled, with no task-specific decoding overrides. After whitespace stripping, empty outputs, `NO REPLY`, and outputs beginning with `[SILENT]` are treated as silence; all other outputs are treated as speech. Silence is canonicalized as `[SILENT]` before being appended to the history.

B.3 Metric Definitions and Aggregation

Reference-based metrics. We measure agreement between predicted and reference speech activity at one-second resolution. For episode i with T_i steps, let $m_{i,t}, g_{i,t} \in \{0, 1\}$ denote model and reference speech activity at step t. Pooling counts across all eligible episodes, we compute

$$\mathrm{F1} = \frac{2 \sum_i \sum_{t=1}^{T_i} m_{i,t} g_{i,t}}{\sum_i \sum_{t=1}^{T_i} (m_{i,t} + g_{i,t})}. \tag{6}$$

Matching is exact, with no temporal tolerance. Episodes without reference speech are excluded; empty predictions contribute only false negatives.

Model-based interaction metrics. We use GPT-5.2 to score three complementary interaction metrics on a 1–10 scale. All metrics' judge prompt are provided in Appendix F.2

- *LiveU (↑).* LiveU measures streaming delivery through timing and turn-taking (*Time*), speech density and pacing (*Rate*), and spoken clarity (*TextU*). It uses the context and timestamped reference and prediction without video frames.
- *FinalQ (↑).* FinalQ measures the quality of the consolidated spoken content through factual consistency (*Fidelity*), discourse coherence (*Continuity*), and informative value (*Substance*). The reference is treated as one valid realization rather than a unique target.
- *MM-TrajQ (↑).* MM-TrajQ measures the quality of the complete temporally aligned multimodal interaction trajectory, including visual grounding, timing, specificity, continuity, usefulness, and partner coordination. It uses the same rubric as the semantic RL reward, but evaluates one trajectory per call and reports the unnormalized score.

C Data Details

C.1 Construction by Interaction Setting

We map all sources into a common format containing the video, duration, and utterances with speaker identities and start/end timestamps. Task queries and speaker personas are included when available. For collaborative settings, we additionally retain the other speaker's utterances as a temporally aligned partner track.

Solo and collaborative commentary. We collect English game and sports broadcasts from Bilibili. We use WhisperX with `large-v3` for transcription, followed by word-level alignment and speaker diarization. We then use gpt-5.2 to correct transcription errors, normalize domain-specific terms, and remove redundant fillers while preserving the original meaning, speaker labels, and timestamps. For each speaker, gpt-5.2 summarizes their speaking style from the transcript, including tone, vocabulary, and pacing. Single-speaker broadcasts provide solo commentary, whereas multi-speaker broadcasts provide collaborative commentary with separate speaker tracks. These tracks preserve the original timing of responses, handoffs, and overlapping speech. Finally, we use gpt-5.2 to shorten utterances that exceed six words per second while retaining their timestamp intervals, and check the resulting samples against this speech-rate limit.

Table 6: Training-data statistics by interaction setting. Commentary aggregates solo and collaborative commentary. OOD evaluation data are excluded, and hours are rounded by row.

	Supervised fine-tuning		Reinforcement learning	
Task	Clips	Hours	Trajectories	Hours
Solo	7,222	72.0	443	5.7
Collaborative commentary	15,000	149.9	885	10.9
Guidance	13,334	203.7	590	6.3
Collaborative presentation	3,600	35.9	74	0.8
Total	39,156	461.3	1,992	23.7

Guidance. We construct guidance from the annotated goals and actions in Ego4D GoalStep, EPIC-KITCHENS, and HoloAssist. For Ego4D GoalStep, we use `gpt-5-chat` to organize the annotations into coherent subtasks, formulate a task query for each subtask, and rewrite action descriptions as concise instructions. The model receives the annotated goals and action intervals and is instructed to preserve their temporal order. For Minecraft, we process visual-only gameplay in five-minute segments using `qwen3-vl-plus` at two frames per second. The model first identifies goals and meaningful task segments, then extracts the observable actions within each segment. We use gpt-5.2 to convert these actions into natural instructions while preserving their timestamps. Across guidance sources, each instruction is shifted to precede its target action, with its endpoint set to the action-onset anchor. We clip the interval at the video boundary and the end of the preceding instruction, and discard instructions with no remaining duration. EgoExoLearn follows the same output format and is used only for OOD evaluation.

Collaborative presentation. We render PDF slide decks and and use gpt-5.2 to generate a two-speaker presentation grounded in their content. Each turn is assigned a speaker and a slide, and the dialogue includes explanations, clarification questions, and handoffs. We vary the speakers' relative contributions to cover both balanced and lead–support presentations. A second model pass checks the dialogue against the slides and revises unsupported content and unnatural exchanges. We synthesize each turn with Edge TTS using distinct voices (`en-US-GuyNeural` and `en-US-JennyNeural`). The measured speech durations determine utterance timestamps and slide display intervals, yielding synchronized presentation videos and dialogue tracks. For each target speaker, the other speaker's utterances form the fixed partner track.

C.2 Scale and Splits

Training data. Table 6 summarizes the training data across the four interaction settings. OpenProact-SFT contains 39,156 clips spanning 461.3 hours, while OpenProact-RL contains 1,992 trajectories spanning 23.7 hours.

Evaluation splits. Table 7 reports the source-level composition of OpenProact-ID and OpenProact-OOD. OpenProact-ID contains 2,000 clips spanning 23.1 hours, with 500 clips per interaction setting, drawn from source domains represented in training. OpenProact-OOD contains 481 clips spanning 5.6 hours from TFT, COA, and EgoExoLearn, which are excluded from both SFT and RL training.

Split construction. When constructing OpenProact-ID, candidate clips that overlap the training set are removed to ensure no data leakage. OpenProact-OOD comprises TFT, COA, and EgoExoLearn, all excluded from the SFT and RL source lists.

D Human Evaluation Protocol

Participants and assignment. We recruited six human annotators for a model-blinded evaluation of three systems: Proact-VL, Proact-VL2$_{\text{SFT}}$, and Proact-VL2$_{\text{RL}}$. We pay an hourly rate of 60 RMB

Table 7: Evaluation split statistics by interaction setting and source. Dashes indicate that a source is not included in that split. Hours are rounded independently; totals use the overall reported durations. Complete-video streaming evaluation is excluded.

		OpenProact-ID		OpenProact-OOD	
Setting	Source	Clips	Hours	Clips	Hours
Solo commentary	Street Fighter	166	2.2	–	–
	MLB	167	2.2	–	–
	Soccer	167	2.2	–	–
	TFT	–	–	200	2.6
Collaborative commentary	CS:GO	84	1.0	–	–
	LoL	83	1.0	–	–
	KPL	83	1.1	–	–
	VCT	83	1.1	–	–
	Delta Force	84	1.1	–	–
	PGC	83	1.1	–	–
	COA	–	–	81	1.1
Guidance	Ego4D GoalStep	125	1.1	–	–
	EPIC-KITCHENS	125	1.2	–	–
	HoloAssist	125	1.2	–	–
	Minecraft	125	1.5	–	–
	EgoExoLearn	–	–	200	1.8
Collaborative presentation	Web slide decks	500	5.2	–	–
Total		2,000	23.1	481	5.6

for annotation. The same 20 video clips were evaluated for all three systems, producing 60 clip–model pairs and 180 ratings overall. Each pair was scored by three distinct annotators. The primary paired comparison between Proact-VL2$_{\text{SFT}}$ and Proact-VL2$_{\text{RL}}$ comprised 40 clip–model pairs and 120 ratings. Assignments were allocated according to annotator availability: two annotators completed 18 items each, while the remaining four completed 36 items each. System identities were replaced with stable labels (*Model A–C*), and outputs were presented individually rather than side by side.

Annotation procedure. For each assigned item, annotators watched the complete video clip with the model output rendered as timestamp-aligned speech and subtitles. The original soundtrack was removed to prevent source commentary from being mistaken for model-generated speech, while model and co-host utterances were visually distinguished. Annotators could inspect the task conditions, including the query, persona, and interaction type, together with a reference transcript presented as one valid trajectory rather than the unique correct response. They then applied a task-specific 1–10 rubric covering visual grounding, factual correctness, temporal alignment, coverage or actionability, and interaction quality. The rubric included explicit score anchors and hard upper bounds for substantial errors or omissions. Annotators assigned an integer score and could optionally record factual errors or other observations in a free-text field. System identities were anonymized, and annotators had no access to MM-TrajQ scores or rationales.

Web interface. We implemented the annotation interface as a local Streamlit application, illustrated in Fig. 4. The main panel contained the video player, synchronized model and co-host subtitles, task context, reference transcript, scoring rubric, and human scoring form. A sidebar displayed the annotator identity, assigned workload, completion progress, and navigation controls. Ratings could be saved and revised, and the *Save and continue* control advanced to the next unfinished item.

Paired SFT–RL comparison. We first averaged the three human ratings for each clip–model pair and then computed the paired difference between Proact-VL2$_{\text{RL}}$ and Proact-VL2$_{\text{SFT}}$ on each of the 20 shared clips. The mean human score increased from 3.87 for SFT to 6.18 for RL, corresponding to a paired improvement of 2.32 points (95% paired-bootstrap CI $[1.63, 3.02]$). The aggregated RL score exceeded the SFT score on 19 of the 20 clips, with no ties, giving a score-implied RL

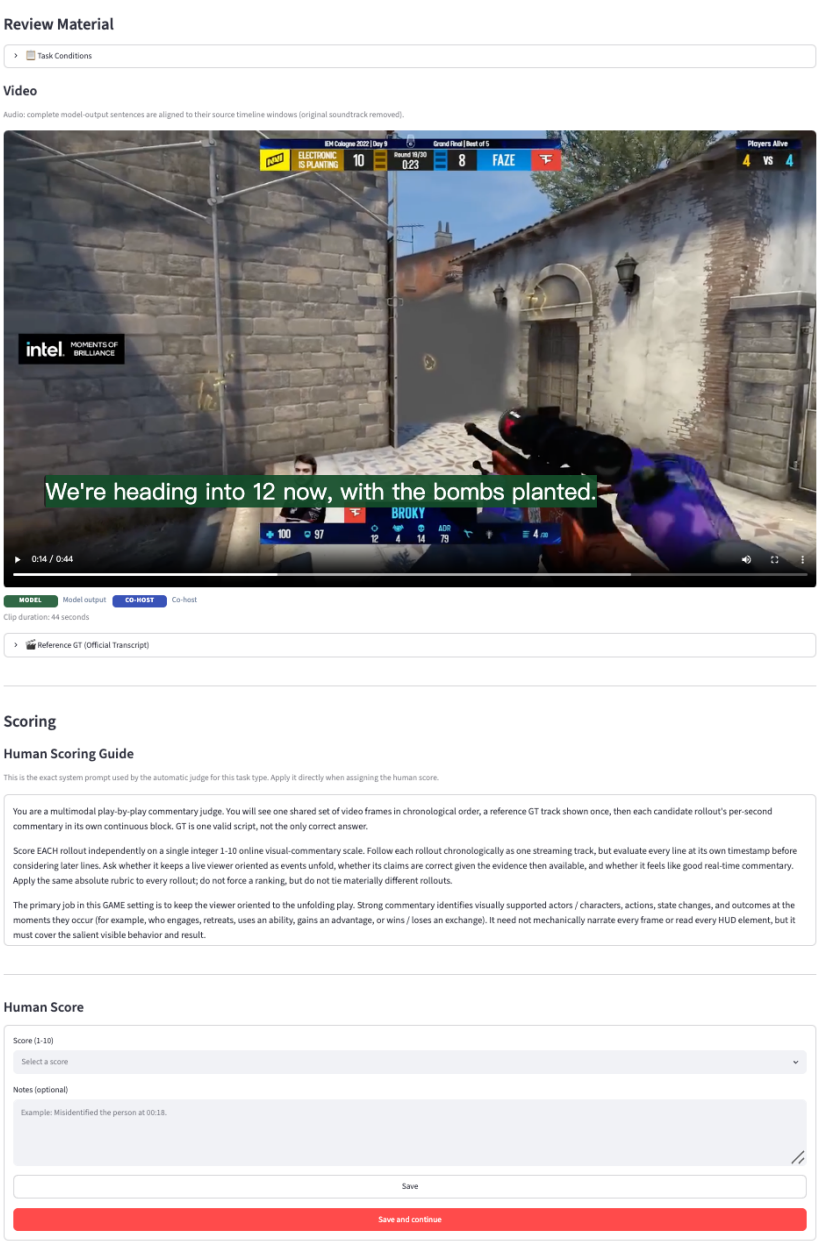


Figure 4: Web interface used for the model-blinded human evaluation of Proact-VL, Proact-VL2$_{\text{SFT}}$, and Proact-VL2$_{\text{RL}}$. Annotators reviewed the video, timestamp-aligned model output, task context, reference transcript, and scoring rubric before assigning a score.

preference rate of 95.0% (Wilson 95% CI $[76.4\%, 99.1\%]$). Because annotators scored outputs separately, this preference rate is derived from the paired quality scores rather than from a direct forced-choice question. Bootstrap uncertainty was estimated by resampling the 20 source clips with replacement for 20,000 iterations.

Agreement analysis. As a secondary validation of MM-TrajQ, we compared its scores with the aggregated human ratings for all 60 clip–model pairs across Proact-VL, Proact-VL2$_{\text{SFT}}$, and Proact-VL2$_{\text{RL}}$. MM-TrajQ achieved Spearman's $\rho = 0.917$ (95% CI $[0.830, 0.965]$) against the mean human score and linearly weighted $\kappa = 0.878$ (95% CI $[0.785, 0.935]$) against the median human

score. Inter-annotator reliability was also substantial, with ordinal Krippendorff's $\alpha = 0.766$ (95% CI $[0.611, 0.870]$).

E Qualitative Study: Failure Annotation

Identification of failure modes. The two failure modes were first identified through human author review of model trajectories and their corresponding interaction context. The first author examined what the models said, when they said it, and how their responses related to the video and, when applicable, the interaction partner's speech. The first author grouped recurring errors into two non-exclusive categories: *multimodal grounding errors*, in which a response makes a factual claim unsupported by or inconsistent with the observed content, and *interaction-state tracking errors*, in which a response misidentifies the scene, action, task, or conversational state relevant at that time. The authors discussed category boundaries and translated these definitions into a common annotation prompt with explicit inclusion and exclusion criteria. These categories provide an exploratory description of observed failures rather than an exhaustive taxonomy of interaction quality.

Annotation process. We used `gpt-5.6-sol` to annotate the model trajectories according to the author-defined categories. For each episode, the annotator received the corresponding video segment, the temporally aligned partner speech when applicable, the assigned role or persona, and the complete timestamped model trajectory, including its utterances and speaking or waiting decisions. These inputs allowed the annotator to inspect both the evidence surrounding an individual response and the sequence of interaction leading to it.

The annotator assessed the supplied material directly and produced its own failure labels and supporting explanations. To ensure the accuracy of the automated labeling, the authors manually reviewed a subset of the initial annotation results and iteratively refined the annotation prompt. The final annotation prompt and category definitions were applied uniformly across models, with the identity of the model producing each trajectory withheld. Each positive annotation was tied to a specific response and the corresponding contextual evidence, allowing the authors to check whether the assigned category was supported.

E.1 Annotation Prompt

Annotation Prompt

Role. You are annotating two types of failure in a streaming interaction trajectory. Assess the trajectory directly against the supplied video, partner speech, and role or persona. Apply the same criteria to every trajectory. Do not infer model identity from writing style, response length, or wording, and do not assume that any model should perform better than another.

Input. For one interaction episode, you receive:

1. The video segment associated with the interaction.
2. The temporally aligned speech of the interaction partner, when the episode includes a partner.
3. The role or persona assigned to the model.
4. The complete model trajectory, with timestamps for its utterances and speaking or waiting decisions.

Use these materials as the evidence for your annotations. No assessment or rationale from another judge is provided.

Inspection procedure. Read the assigned role and inspect the full trajectory together with the video and partner speech. For each candidate failure, identify the relevant response, its timestamp, and the evidence needed to interpret it. Consider preceding responses and partner turns when determining what the model is referring to and whether the interaction state has changed.

Assess each response against the information available up to its timestamp. Do not use a later event to justify a factual claim that was unsupported when made, or to penalize a response for failing to anticipate that event. Distinguish descriptions of current events, references to earlier events, predictions, and prospective instructions.

Unit of annotation. Annotate a specific claim or response span within the trajectory. Use surrounding turns as context when necessary, but identify the particular span that supports each label. The two labels

Annotation Prompt (continued)

are non-exclusive: a span may receive neither label, either label, or both labels. If both apply, explain why each definition is satisfied. Record the same response span at most once under each label.

Label A: Multimodal grounding error. Assign this label when the model presents a concrete factual claim about an event, action, object, identity, attribute, quantity, or state that is unsupported by or contradicted by the supplied multimodal evidence. Identify the claim and explain which observable evidence contradicts it or why the available evidence does not support the asserted detail.

Do not label a response solely because it is vague, omits relevant information, or describes a genuine earlier event with a delay. A suggestion or instruction is not a grounding error merely because the proposed action has not yet occurred. Likewise, a clearly qualified prediction is not a factual assertion that the predicted event has already happened. However, a suggestion or prediction may contain a separate unsupported factual premise, which should be assessed on its own.

Limited visibility or ambiguous speech does not by itself establish that a claim is false. When the supplied material is insufficient to distinguish an error from a plausible interpretation, do not force a positive label.

Label B: Interaction-state tracking error. Assign this label when a response is based on a specific incorrect identification of the interaction state at its timestamp. This includes confusing the current scene, action, task, participant, topic, or task stage, or continuing to respond as though an earlier state remains current after the supplied context establishes a transition.

Identify both the state implied by the response and the state supported by the video or partner speech. When the error involves a transition, describe the relevant change and explain how the response fails to account for it.

Do not label generic vagueness, low usefulness, omission, or awkward turn-taking without a specific state mismatch. A retrospective comment about an earlier event is not a tracking error merely because the scene has since changed. Similarly, a delayed but correctly framed response is not sufficient evidence of a tracking error. Assess references to earlier events in context rather than assuming that every utterance describes the present moment.

Evidence and neutrality. For every positive label, quote the relevant model response and describe the supporting video or partner-speech evidence with timestamps. Write your own explanation from these inputs. Do not substitute general criticism for a demonstrable instance of either failure mode. Apply the same evidentiary threshold to every trajectory, including responses that otherwise appear fluent or high quality. Do not seek a particular number of failures, an equal number across trajectories, or a preferred comparison outcome.

Output. Return one list for each failure mode. Each entry must contain the response timestamp or interval, the verbatim response span, the relevant contextual evidence, and an explanation of why the label applies. Use an empty list when no supported instance is identified.

```
{
  "multimodal_grounding_error": [
    {
      "timestamp": "<response time or interval>",
      "response": "<verbatim response span>",
      "evidence": "<contextual evidence with timestamps>",
      "explanation": "<why Label A applies>"
    }
  ],
  "interaction_state_tracking_error": [
    {
      "timestamp": "<response time or interval>",
      "response": "<verbatim response span>",
      "evidence": "<contextual evidence with timestamps>",
      "explanation": "<why Label B applies>"
    }
  ]
}
```

E.2 Aggregation and Audit

The authors reviewed the assigned labels against the corresponding response spans and multimodal evidence to check their consistency with the category definitions. Ambiguous cases were discussed

using the same criteria across models. An episode was counted as positive for a failure mode if at least one supporting instance was retained after review. Multiple instances of the same failure mode within an episode contributed only once to its episode-level indicator. Because the categories are non-exclusive, an episode could be positive for both modes.

For each model and failure mode, we computed the percentage of positive episodes over the same set of matched events across all three models. Thus, the percentages describe the occurrence of each failure mode at the episode level, rather than the number of erroneous utterances or their severity.

The common criteria, withheld model identities during automated annotation, and review of supporting evidence were intended to reduce differential treatment across models. Nevertheless, the categories originated from author inspection, and the final labels involved both automated interpretation and author judgment. Accordingly, we treat these results as descriptive findings within the analyzed episodes. They do not establish an exhaustive error rate or constitute an independent human evaluation of overall interaction quality. An episode without a retained label indicates that neither specified failure was identified under this procedure; it does not establish that the trajectory is free of other errors.

E.3 Diversity Metrics

We measured lexical diversity using the same aligned examples across all three models. For each response, we concatenated its active timestamped utterances in chronological order, excluding empty entries, explicit silence markers, and continuation placeholders. We lowercased the resulting word tokens before computing either metric.

MATTR. We computed the moving-average type–token ratio using a window of 25 tokens. For each response containing at least 25 tokens, we calculated the fraction of unique tokens in every contiguous 25-token window and averaged these fractions. The reported value for each model is the mean over its eligible responses. Higher MATTR indicates greater local lexical variety within a response; responses shorter than 25 tokens do not contribute to this metric.

Self-BLEU. We computed Self-BLEU using uniform weights over 1–4-grams. Each non-empty response served as the hypothesis, with all other responses from the same model, interaction type, and dataset serving as references. We used maximum reference counts for modified n-gram precision, method-1 smoothing with $\epsilon = 0.1$ for zero numerators, and the standard brevity penalty. The reported value for each model is the mean over responses. Lower Self-BLEU indicates less lexical overlap among outputs across episodes.

For both metrics, error bars represent 95% percentile intervals from 5,000 deterministic episode-level bootstrap replicates. These metrics characterize lexical variation within responses and lexical similarity across outputs, respectively. They do not by themselves measure semantic diversity, response correctness, or variation across repeated generations for the same input.

F Exact Prompts and Output Schemas

F.1 Policy and Baseline-Adapter Prompts

Policy system-prompt assembly

[[Task]]
<SOURCE_DESCRIPTION> <TASK_INSTRUCTION>
[[Persona]]
<PERSONA>
[[Output]]
If nothing should be said at this moment, output exactly [SILENT] and nothing else. Otherwise, speak naturally.

Solo commentary: task instruction

Your role is to independently analyze and narrate the game, delivering insightful, engaging, and natural commentary just like a human expert. Focus on key plays, tactics, player actions, and exciting moments to keep viewers informed and entertained. It is not necessary to speak continuously—during uneventful or transitional parts of the match, you may remain silent. Always maintain a lively yet professional tone, and adapt your commentary to the real-time action shown in the video. Please always use polite, restrained, and family-friendly language, and do not use any profanity, insults, or discriminatory slurs.

Collaborative commentary: task instruction

Working alongside a human co-caster in a live broadcasting scenario, your role is to analyze, interpret, and explain the in-game action, highlight exciting plays, and engage viewers with insightful and entertaining commentary. You should respond naturally to your co-caster's remarks, support their analysis, or introduce new perspectives, just like a professional esports commentator team. Always keep your tone lively, professional, and audience-friendly. Rely on real-time video and your co-caster's speech to guide your commentary, and make sure your responses are timely, relevant, and complementary to your co-caster. Please always use polite, restrained, and family-friendly language, and do not use any profanity, insults, or discriminatory slurs.

Guidance: task instruction

Use the real-time visual scene to provide clear, step-by-step guidance that helps the user accomplish their current goal. Speak only when there is something genuinely useful to say — for example, when the user is starting a new sub-action, has just finished one and needs direction for the next, or is about to make a mistake. Otherwise, stay silent. Keep each instruction concise, accurate, and grounded in what is visible right now. Please always use polite, restrained, and family-friendly language, and do not use any profanity, insults, or discriminatory slurs.

Collaborative presentation: task instruction

Alongside your human co-presenter, you are giving this talk from your own slides on screen. Present each slide in the first person — walk the audience through your text, numbers, figures, and equations ("here we've got…", "as you can see…") and connect the ideas across slides. Respond naturally to your co-presenter's points: confirm, expand, push back, or hand the floor back. Speak when the current slide warrants it; otherwise let your co-presenter carry the moment. Keep the tone friendly, lively, and educational, own the material as a presenter (never "the deck"/"the slide" in the third person), and stay grounded in what is actually on the current slide. Please always use polite, restrained, and family-friendly language, and do not use any profanity, insults, or discriminatory slurs.

F.2 LIVEU AND FINALQ PROMPTS

LiveU: Streaming Delivery

Evaluate only the model's timestamped prediction, using the context and reference timeline as supporting text. `[ASSISTANT]` identifies the model, `[USER]` a user question, and `[SPEAKER*]` another speaker. Missing or explicitly silent steps represent silence. The reference supports assessment of speaking opportunities and consistency with key events, without requiring identical wording or exact temporal boundaries.

Dimensions. *Time*: timely responses, coverage of relevant speaking opportunities, appropriate silence, and turn-taking. *Rate*: listenable speech density and pacing without dense information dumps or repetitive filler. *TextU*: clear, natural spoken wording that is easy to follow in real time.

Task-specific criteria.

– *Solo and collaborative commentary:* react within key-event windows and favor short, event-paced contributions. Brief one-second overlap can be acceptable; sustained (two or more consecutive seconds), frequent, or disruptive overlap is a defect.

LiveU: Streaming Delivery (continued)

- *Guidance:* respond to questions, hesitation, errors, and task transitions with actionable, stepwise instructions. Allow silence while the user executes a step that needs no verbal cue.
- *Collaborative presentation:* follow slide transitions and partner handoffs. Longer narration is acceptable when listenable and aligned with the current slide. Brief overlap of less than one second or a back-channel is acceptable; sustained or repeated overlap is a defect.

Score anchors. Apply separately to each dimension: **10**, excellent; **7–9**, good with minor issues; **4–6**, noticeable timing, pacing, or clarity problems; **2–3**, severe problems but limited usability; **1**, unusable. Reasonable shifts within overlapping speaking windows are acceptable. Expressive language is allowed when consistent with events and not disruptive. Mild fragmentation can remain usable, but fragmentation that harms pacing or clarity lowers the corresponding score.

Inputs. Context; reference timeline; prediction timeline.

Output. Return only three integer scores, one per line:

```
Time:  <1--10>
Rate:  <1--10>
TextU: <1--10>
```

FinalQ: Completed Interaction Quality

Evaluate the model's consolidated prediction in chronological order, using the context, query, and reference as supporting text. The reference is one valid realization: different wording, valid steps, or emphasis can receive full credit. Changing actions or claims at different moments is not itself a contradiction; assess incompatibility at the same moment, task state, or slide.

Dimensions. *Fidelity*: temporal consistency and freedom from fabricated or misleading details. *Continuity*: coherent progression, consistent references, and appropriate build-on from context or partner turns. *Substance*: specific, useful information beyond repetition, generic filler, or acknowledgement.

Task-specific criteria.

- *Solo and collaborative commentary:* assess plausible, on-topic narration, consistency with the implied event sequence, and informative contributions that complement prior commentary.
- *Guidance:* assess whether the instructions plausibly complete the query safely and correctly. Alternative valid procedures are allowed; wrong tools, harmful actions, and omitted genuinely required steps are defects. A clearly required omission lowers Fidelity and Substance to 5–6 or below. Opposite instructions at different procedural stages need not conflict.
- *Collaborative presentation:* assess consistency with the implied slides, partner-aware transitions, and explanatory value. Normal handoff gaps and first-person-plural presenter language are valid. Invented results, same-slide contradictions, off-deck discussion, and empty agreement are defects.

Score anchors. Apply separately to each dimension: **10**, excellent; **8–9**, strong; **7**, the default for relevant, coherent, readable output with plausible content; **5–6**, identifiable defects with some remaining usefulness; **3–4**, multiple substantial defects; **1–2**, unusable, fundamentally misleading, or harmful. Brevity or divergence from the reference alone does not justify a low score.

Inputs. Context; query; consolidated reference and prediction; optional partner script. The consolidated scripts retain chronological order but omit per-second timestamps. No separate persona field or video frames are supplied.

Output. Return three integer scores and a reason:

```
Fidelity:  <1--10>
Continuity:  <1--10>
Substance:  <1--10>
Reason:  <one sentence, at most 25 words>
```

The reason identifies the concrete strength or defect driving the lowest sub-score.

F.3 MM-TrajQ Reward and Evaluation Prompts

MM-TrajQ: Multimodal Trajectory Quality

Score each rollout independently on an integer 1–10 scale, using the same absolute rubric without forcing a ranking. The reference is one valid realization, not a unique target. Evaluate the temporally aligned trajectory while checking each statement against evidence available when it was spoken. Later frames may establish the event sequence and action correspondence, but cannot retrospectively ground an earlier claim or recover a missed speaking opportunity.

Shared criteria. Required task coverage and factual grounding come before delivery quality. Silence is appropriate when no useful contribution is needed or the partner holds the floor; silence through required opportunities is an omission. Reward supported specificity, continuity, timely contributions, and useful explanation or emotion. Judge persona cadence over the clip, without imposing a fixed speaking quota. Short fragments spanning consecutive seconds are not inherently defects. When a partner track is present, opening a turn while the partner's line ends in an ellipsis is treated as interruption.

Task-specific criteria.

- *Solo and collaborative commentary:* cover meaningful visible actors, actions, state changes, and outcomes as they unfold. Allow roughly one second of streaming latency within the same event stage; an outcome-stage recap cannot recover earlier setup or action coverage. Grounded analysis supplements play-by-play. Anticipation is optional and requires a visible action-specific cue; movement or proximity alone does not establish intent. A qualified uncertain prediction is not wrong merely because its outcome does not occur. For scores 8–10, roughly 60% or more of substantive commentary during active play should describe observed events, with timely calls for at least half of eligible setup/commitment sequences; scores 9–10 require roughly 80% setup/commitment coverage. Do not apply this ratio when no such cues are visible.
- *Guidance:* give concrete, actionable instructions grounded in the visible task state. Proactive credit requires delivery before the action's earliest purposeful onset, including reaching, grasping, or positioning. Later matching behavior establishes correspondence, not causality. A useful correction during an action earns credit for the new adjustment. Cover consequential guidance opportunities while allowing execution silence. Progress, readiness, completion, and next-step claims require evidence already visible; hedging or eventual success does not justify an unsupported state claim.
- *Collaborative presentation:* require all three of accurate slide content, substantive partner collaboration, and original grounded information. A substantive turn must identify a concrete slide item; a broad topic mention is insufficient on a content-rich slide. Interpretation must add information beyond paraphrase or praise. Credit each grounded claim separately: one valid anchor does not justify generic filler later in the turn. Acknowledgement or polite silence alone earns no collaboration credit, and partner coverage does not replace the model's own substantive contribution.

Score caps.

- *Shared:* a localized factual error caps the score at 7; a central contradiction at 4; repeated unsupported or contradicted specifics warrant 1–3. Material cadence mismatch caps the score at 6, and severe persistent mismatch at 4.
- *Commentary:* missing more than half of eligible setup/commitment moments caps the score at 7; missing at least 80%, or repeatedly reporting only confirmed outcomes, caps it at 6. Predominantly generic truths cap it at 5. Analysis outweighing concrete play-by-play during salient events caps it at 6, or 4 when decisive beats are repeatedly missed. Predominantly waiting and recapping caps it at 4; no live coverage of decisive beats caps it at 3.
- *Guidance:* repeatedly speaking after action onset caps the score at 4; exclusively echoing already-started or completed actions caps it at 3. Mostly non-actionable instructions cap it at 4; entirely generic instructions at 3. Missing more than half of multiple clear opportunities caps it at 5; missing all, or effective silence despite consequential opportunities, at 2. A central unsupported progress or completion claim caps it at 4; repeated claims warrant 1–3.
- *Presentation:* absent concrete slide content, predominantly unanchored or generic turns, repeated acknowledgement farming, a serious slide-fact contradiction, or repeated partner interruption/neglect caps the score at 4. Adequate slide description without meaningful collaboration or grounded interpretation caps it at 6; all three dimensions present but one clearly thin caps it at 7. Complete silence despite natural handoffs caps it at 2.

Score anchors. Subject to the applicable caps: **10**, exceptional coverage, grounding, timing, and delivery; **8–9**, strong with limited defects; **7**, good but uneven; **5–6**, partial or mixed quality; **3–4**, major failures; **1–2**, nearly unusable or a complete failure.

MM-TrajQ: Multimodal Trajectory Quality (continued)

Inputs. Chronological frames; task query and persona; reference trajectory; per-second candidate trajectories; available partner speech. The textual input precedes the shared frame-label/image sequence. Training can additionally supply a structured cadence target; evaluation does not append this separate field.

Output. Return JSON with a `scores` array containing one object per candidate id. Each object has the following fields:

```
{"id": "<ID>", "MM-TrajQ_score": <integer 1--10>,
 "reason": "<grounded rationale>",
 "suggestion": "Overall strategy: <policy>"}
```

Each reason is approximately 60–80 words and cites decisive visible moments, coverage, and grounding. Each suggestion is one or two sentences, at most 60 words including the required prefix, and states one transferable causal policy using evidence observed so far. It must not retell a timestamped incident, list local fixes, or impose a rigid speaking quota.